Galina Setlak, Mikhail Alexandrov,
Krassimir Markov
(editors)

Artificial Intelligence Methods and Techniques for Business and Engineering Applications

I T H E A®
Rzeszow - Sofia
2012

Galina Setlak, Mikhail Alexandrov, Krassimir Markov (Eds.)

Artificial Intelligence Methods and Techniques for Business and Engineering Applications

ITHEA®

2012, Rzeszow, Poland; Sofia, Bulgaria,

ISBN: 978-954-16-0057-3 (printed)
ISBN: 978-954-16-0058-0 (online)

ITHEA IBS ISC No.: 26

First edition

Printed in Poland

Recommended for publication by The Scientific Concil of the Institute of Information Theories and Applications FOI ITHEA

This issue contains a monograph that concern actual problems of research and application of information technologies, especially the new approaches, models, algorithms and methods for information modeling to be used in business and engineering applications of intelligent and information systems.

It is represented that book articles will be interesting for experts in the field of information technologies as well as for practical users.

ISBN: 978-954-16-0057-3 (printed)
ISBN: 978-954-16-0058-0 (online)

C\O JUSAUTOR, SOFIA, 2012

PREFACE

This monograph presents the current problems of research and applications of Information and Communication Technologies, with the special focus on the new approaches developed in the area of widely understood Artificial Intelligence (AI). AI is one of the newest fields in science and engineering and is relevant to any intellectual task. It is universal and at the same time big field that encompasses a huge variety of subfields, ranging from automatic control in engineering systems driven by intelligent algorithms to the AI techniques and methods applications to supporting decision making in contemporary organizations. What can AI do today? A short and concise answer is very difficult because of many activities in many subfields. Therefore we hope that this collection of papers will provide all interested researchers and systems developers with a fresh view of how the modern AI applications to Business and Engineering problems look like. To this end, the following chapters include the papers describing models, algorithms and methods of AI used in Information and Communication Technologies applications.

Main topics which are covered in the issue are:

- Automatic Control Systems
- Natural Language Processing and Web Mining
- Intelligent Agents and Multi-Agent Systems
- Artificial Intelligence in Modeling and Simulation
- Business Intelligence Systems
- Neural Networks, Machine Learning
- Bioinformatics using Intelligent and Machine Learning
- Decision Making Support and Expert Systems

We express our thanks to all authors of this collection as well as to all who support its publishing

Rzeszow – Sofia *G. Setlak, M.Alexandrov, K. Markov*
September 2012

TABLE OF CONTENTS

BUSINESS INTELLIGENCE SYSTEMS

NEURAL NETWORKS, MACHINE LEARNING

BIOINFORMATICS USING INTELLIGENT AND MACHINE LEARNING

DECISION MAKING SUPPORT AND EXPERT SYSTEMS

INDEX OF AUTHORS

AUTOMATIC CONTROL SYSTEMS

POSITIVE STABLE REALIZATIONS OF CONTINUOUS-TIME LINEAR SYSTEMS

Tadeusz Kaczorek

Abstract: *The problem for existence and determination of the set of positive asymptotically stable realizations of a proper transfer function of linear continuous-time systems is formulated and solved. Necessary and sufficient conditions for existence of the set of the realizations are established. Procedure for computation of the set of realizations are proposed and illustrated by numerical examples.*

Keywords: *positive, stable, realization, existence, procedure, linear, continuous-time, system.*

Introduction

Determination of the state space equations for given transfer matrix is a classical problem, called realization problem, which has been addressed in many papers and books [Farina and Rinaldi 2000, Benvenuti and Farina 2004, Kaczorek 1992, 2009b, 2011c, 2012, Shaker and Dixon 1977]. An overview on the positive realization problem is given in [Farina and Rinaldi 2000, Kaczorek 2002, Benvenuti and Farina 2004]. The realization problem for positive continuous-time and discrete-time linear systems has been considered in [Kaczorek 2004, 2006a, 2006b, 2006c, 2011a, 2011b, 2011c] and the positive realization problem for discrete-time systems with delays in [Kaczorek 2004, 2005, 2006c]. The fractional positive linear systems has been addressed in [Kaczorek 2008a, 2009a, 2011c]. The realization problem for fractional linear systems has been analyzed in [Kaczorek 2008b] and for positive 2D hybrid systems in [Kaczorek 2008c]. A method based on similarity transformation of the standard realization to the discrete positive one has been proposed in [Kaczorek 2011c]. Conditions for the existence of positive stable realization with system Metzler matrix for transfer function has been established in [Kaczorek 2011a]. The problem of the existence and determination of the

set of Metzler matrices for given stable polynomials has been formulated and solved in [Kaczorek 2012].

It is well-known that [Farina and Rinaldi 2000, Kaczorek 1992, 2002] that to find a realization for a given transfer function first we have to find a state matrix for given denominator of the transfer function.

In this paper necessary and sufficient conditions for existence of the set of positive stable realizations of a proper transfer function of linear continuous-time systems are established and a procedure for computation of the set of realizations is proposed.

The paper is organized as follows. In section 2 some preliminaries concerning positive linear systems are recalled and the problem formulation is given. Problem solution for systems with real negative poles of the transfer function is presented in section 3. The problem of the existence and computation of the set of positive asymptotically stable realizations for systems with complex conjugate poles is addressed in section 4. Concluding remarks are given in section 5.

The following notation will be used: $\Re$ - the set of real numbers, $\Re^{n\times m}$ - the set of $n\times m$ real matrices, $\Re_+^{n\times m}$ - the set of $n\times m$ matrices with nonnegative entries and $\Re_+^n = \Re_+^{n\times 1}$, M_n - the set of $n\times n$ Metzler matrices (real matrices with nonnegative off-diagonal entries), M_{ns} - the set of $n\times n$ asymptotically stable Metzler matrices, I_n - the $n\times n$ identity matrix, A^T - transpose of the matrix A, $\Re^{n\times m}(s)$ - the set of $n\times m$ rational matrices in s.

Preliminaries and the problem formulation

Consider the continuous-time linear system

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{2.1a}$$

$$y(t) = Cx(t) + Du(t) \tag{2.1b}$$

where $x(t)\in\Re^n$, $u(t)\in\Re^m$, $y(t)\in\Re^p$ are the state, input and output vectors and $A\in\Re^{n\times n}$, $B\in\Re^{n\times m}$, $C\in\Re^{p\times n}$, $D\in\Re^{p\times m}$.

Definition 2.1. [Farina and Rinaldi 2000, Kaczorek 2002] The system (2.1) is called (internally) positive if $x(t)\in\Re_+^n$, $y(t)\in\Re_+^p$, $t\geq 0$ for any initial conditions $x(0)=x_0\in\Re_+^n$ and all inputs $u(t)\in\Re_+^m$, $t\geq 0$.

Theorem 2.1. [Farina and Rinaldi 2000, Kaczorek 2002] The system (2.1) is positive if and only if

$$A \in M_n,\ B \in \Re_+^{n \times m},\ C \in \Re_+^{p \times n},\ D \in \Re_+^{p \times m}. \tag{2.2}$$

Definition 2.2. [Farina and Rinaldi 2000, Kaczorek 2002] The positive system (2.1) is called asymptotically stable if

$$\lim_{t \to \infty} x(t) = 0 \text{ for any } x_0 \in \Re_+^n. \tag{2.3}$$

Theorem 2.2. [Farina and Rinaldi 2000, Kaczorek 2002]The positive system (2.1) is asymptotically stable if and only if all coefficients of the polynomial

$$p_n(s) = \det[I_n s - A] = s^n + a_{n-1}s^{n-1} + ... + a_1 s + a_0 \tag{2.4}$$

are positive, i.e. $a_i > 0$ for $i = 0,1,...,n-1$.

Definition 2.3. [Kaczorek 2002] A matrix $P \in \Re_+^{n \times n}$ is called the monomial matrix (or generalized permutation matrix) if its every row and its every column contains only one positive entry and its remaining entries are zero.

Lemma 2.1. [Kaczorek 2002] The inverse matrix A^{-1} of the monomial matrix A is equal to the transpose matrix in which every nonzero entry is replaced by its inverse.

Lemma 2.2. If $A_M \in M_n$ then $\overline{A}_M = PA_MP^{-1} \in M_n$ for every monomial matrices $P \in \Re_+^{n \times n}$ and

$$\det[I_n s - \overline{A}_M] = \det[I_n s - A_M]. \tag{2.5}$$

Proof. By Lemma 2.1 if $P \in \Re_+^{n \times n}$ then $P^{-1} \in \Re_+^{n \times n}$ and $\overline{A}_M = PA_MP^{-1} \in M_n$ if $A_M \in M_n$. It is easy to check that

$$\begin{aligned} \det[I_n s - \overline{A}_M] &= \det[I_n s - PA_MP^{-1}] = \det\{P[I_n s - A_M]P^{-1}\} \\ &= \det P \det[I_n s - A_M] \det P^{-1} = \det[I_n s - A_M] \end{aligned} \tag{2.6}$$

since $\det P \det P^{-1} = 1$.

The transfer matrix of the systems (2.1) is given by

$$T(s) = C[I_n s - A]B + D. \tag{2.7}$$

The transfer matrix is called proper if

$$\lim_{s \to \infty} T(s) = K \in \Re^{p \times m} \tag{2.8}$$

and it is called strictly proper if $K = 0$.

Definition 2.4. Matrices (2.2) are called a positive realization of transfer matrix $T(s)$ if they satisfy the equality (2.7).

The realization is called asymptotically stable if the matrix A is an asymptotically stable Metzler matrix (Hurwitz Metzler matrix).

Theorem 2.3. [Kaczorek 2002] The positive realization (2.2) is asymptotically stable if and only if all coefficients of the polynomial

$$p_A(s) = \det[I_n s - A] = s^n + a_{n-1}s^{n-1} + ... + a_1 s + a_0 \tag{2.9}$$

are positive, i.e. $a_i > 0$ for $i = 0,1,...,n-1$.

Lemma 2.3. The matrices

$$\overline{A}_k = PA_kP^{-1} \in M_{ns},\ \overline{B}_k = PB_k \in \Re_+^{n\times m},\ \overline{C}_k = C_kP^{-1} \in \Re_+^{p\times n},\ \overline{D}_k = D_k \in \Re_+^{p\times m},\ k = 1,...,N \tag{2.10}$$

are a positive asymptotically stable realization of the proper transfer matrix $T(s) \in \Re^{p\times m}(s)$ for any monomial matrix $P \in \Re_+^{n\times n}$ if and only if the matrices

$$A_k \in M_{ns},\ B_k \in \Re_+^{n\times m},\ C_k \in \Re_+^{p\times n},\ D_k \in \Re_+^{p\times m},\ k = 1,...,N \tag{2.11}$$

are a positive asymptotically stable realization of $T(s) \in \Re^{p\times m}(s)$.

Proof. By Lemma 2.1 if P is a monomial matrix then $P^{-1} \in \Re_+^{n\times n}$ is also monomial matrix. Hence $\overline{A}_k \in M_{ns}$ if and only if $A_k \in M_{ns}$, $\overline{B}_k \in \Re_+^{n\times m}$ if and only if $B_k \in \Re_+^{n\times m}$ and $\overline{C}_k \in \Re_+^{p\times n}$ if and only if $C_k \in \Re_+^{p\times n}$.

Using (2.10) we obtain

$$\begin{aligned}\overline{T}(s) &= \overline{C}_k[I_n s - \overline{A}_k]^{-1}\overline{B}_k + \overline{D}_k = C_kP^{-1}[I_n s - PA_kP^{-1}]^{-1}PB_k + D_k \\ &= C_kP^{-1}\{P[I_n s - A_k]P^{-1}\}^{-1}PB_k + D_k = C_kP^{-1}P[I_n s - A_k]^{-1}P^{-1}PB_k + D_k \\ &= C_k[I_n s - A_k]^{-1}B_k + D_k = T(s).\end{aligned} \tag{2.12}$$

Therefore, the matrices (2.10) are a positive asymptotically stable realization of $T(s)$ if and only if the matrices (2.11) are also its positive asymptotically stable realization.

The problem under considerations can be stated as follows: Given a rational proper matrix $T(s) \subset \Re^{p\times m}(s)$, find a set of its positive asymptotically stable realizations (2.11).

In this paper necessary and sufficient conditions for existence of the set of the positive asymptotically stable realizations for a given $T(s)$ will be established and a procedure for computation of the set of realizations will be proposed.

Systems with real negative poles

In this section the single-input single-output linear continuous-time linear systems with the proper transfer function

$$T(s) = \frac{b_n s^n + b_{n-1} s^{n-1} + ... + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + ... + a_1 s + a_0} \tag{3.1}$$

having only real negative poles (not necessarily distinct) $-\alpha_1, -\alpha_2, \ldots, -\alpha_n$, i.e.

$$\begin{aligned} &p_n(s) = (s+\alpha_1)(s+\alpha_2)...(s+\alpha_n) = s^n + a_{n-1}s^{n-1} + ... + a_1 s + a_0, \\ &a_{n-1} = \alpha_1 + \alpha_2 + ... + \alpha_n, \quad a_{n-2} = \alpha_1(\alpha_2 + \alpha_3 + ... + \alpha_n) + \alpha_2(\alpha_3 + \alpha_4 + ... + \alpha_n) + ... + \alpha_{n-1}\alpha_n, ..., \\ &a_0 = \alpha_1\alpha_2...\alpha_n \end{aligned} \tag{3.2}$$

will be considered.

First we shall address the problem for $n = 1$ with the transfer function

$$T(s) = \frac{b_1 s + b_0}{s + a}, \; a > 0. \tag{3.3}$$

Theorem 3.1. There exists the set of positive asymptotically stable realizations

$$\overline{A}_k = PA_kP^{-1}, \; \overline{B}_k = PB_k, \; \overline{C}_k = C_kP^{-1}, \; \overline{D}_k = D_k, \; k = 1,2 \tag{3.4}$$

for any positive parameter $P > 0$ and A_k, B_k, C_k, D_k having one of the forms

$$A_1 = [-a], \;\; B_1 = [1], \;\; C_1 = [b_0 - ab_1], \;\; D_1 = [b_1] \tag{3.5}$$

or

$$A_2 = [-a], \;\; B_2 = [b_0 - ab_1], \;\; C_2 = [1], \;\; D_2 = [b_1] \tag{3.6}$$

of the transfer function (3.3) if and only if

$$a > 0, \;\; b_1 \geq 0, \;\; b_0 - ab_1 \geq 0. \tag{3.7}$$

Proof. It is easy to check that the matrices (3.5) are a realization of (3.3). The matrix $A_1 \in M_{1s}$ and $C_1 \in \Re_+^{1\times 1}$, $D_1 \in \Re_+^{1\times 1}$ if and only if the conditions (3.7) are satisfied. By Lemma 2.3 the matrices (3.4) are a positive asymptotically stable realization of (3.3) for any $P > 0$ if and only if the matrices (3.5) are its positive asymptotically stable realization. Proof for matrices (3.6) is similar.

Theorem 3.2. There exists the set of positive asymptotically stable realizations

$$\overline{A}_{Mk} = PA_{Mk}P^{-1} \in M_{2s}, \; \overline{B}_k = PB_k \in \Re_+^{2\times 1}, \; \overline{C}_k = C_kP^{-1} \in \Re_+^{1\times 2}, \; \overline{D}_k = D_k \in \Re_+^{1\times 1}, \quad k = 1,2 \tag{3.8}$$

for any monomial matrix $P \in \Re_+^{2\times 2}$ and A_{Mk}, B_k, C_k, D_k having one of the forms

$$A_{M1} = \begin{bmatrix} -a & a_1 a - a^2 - a_0 \\ 1 & a - a_1 \end{bmatrix}, \quad B_1 = \begin{bmatrix} b_0 - a b_1 + (a a_1 - a_0) b_2 \\ b_1 - a_1 b_2 \end{bmatrix}, \quad C_1^T = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad D_1 = [b_2],$$

$$0 < a < a_1, \quad a_1 a - a^2 - a_0 \geq 0 \tag{3.9a}$$

$$A_{M2} = A_{M1}^T = \begin{bmatrix} -a & 1 \\ a_1 a - a^2 - a_0 & a - a_1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C_2^T = \begin{bmatrix} b_0 - a b_1 + (a a_1 - a_0) b_2 \\ b_1 - a_1 b_2 \end{bmatrix}, \quad D_2 = [b_2],$$

$$0 < a < a_1, \quad a_1 a - a^2 - a_0 \geq 0 \tag{3.9b}$$

of the transfer function

$$T(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^2 + a_1 s + a_0} \tag{3.10}$$

if and only if

$$a_1^2 - 4a_0 \geq 0 \tag{3.11}$$

and

$$b_2 \geq 0, \quad b_0 - a b_1 + (a a_1 - a_0) b_2 \geq 0, \quad b_1 - a_1 b_2 \geq 0 \text{ for } 0 < a < a_1. \tag{3.12}$$

Proof. The matrix $A_{M1} \in M_{2s}$ if and only if its characteristic polynomial

$$\det[I_2 s - A_{M1}] = \begin{vmatrix} s + a & a^2 + a_0 - a_1 a \\ -1 & s + a_1 - a \end{vmatrix} = s^2 + a_1 s + a_0$$

has negative real zeros and this is the case if and only if the condition (3.11) is met and $0 < a < a_1$. The matrix

$$D_1 = \lim_{s \to \infty} T(s) = [b_2] \in \Re_+^{1\times 1}$$

if and only if $b_2 \geq 0$. The strictly proper transfer function has the form

$$T_{sp}(s) = T(s) - D_1 = \frac{\bar{b}_1 s + \bar{b}_0}{s^2 + a_1 s + a_0} \tag{3.13}$$

where $\bar{b}_1 = b_1 - a_1 b_2$, $\bar{b}_0 = b_0 - a_0 b_2$. Assuming C_1 = [0 1] we obtain

$$T_{sp}(s) = C_1[I_2 s - A_{M1}]^{-1} B_1 = [0 \quad 1] \begin{bmatrix} s+a & a^2 + a_0 - a_1 a \\ -1 & s + a_1 - a \end{bmatrix}^{-1} \begin{bmatrix} b_{11} \\ b_{12} \end{bmatrix} = \frac{b_{12} s + b_{11} + a b_{12}}{s^2 + a_1 s + a_0}. \tag{3.14}$$

From comparison of (3.13) and (3.14) we have

$$b_{12} = \bar{b}_1 = b_1 - a_1 b_2,$$
$$b_{11} = \bar{b}_0 - a b_{12} = b_0 - a_0 b_2 - a(b_1 - a_1 b_2) = b_0 - a b_1 + (a a_1 - a_0) b_2. \tag{3.15}$$

From (3.15) it follows that $B_1 \in \Re_+^{2\times 1}$ if and only if the conditions (3.12) are satisfied. The proof for (3.9b) is similar. By Lemma 2.3 the matrices (3.8) are a positive asymptotically stable realization for any monomial matrix $P \in \Re_+^{2\times 2}$ if and only if the matrices (3.9) are its positive asymptotically stable realization.

Example 3.1. Compute the set of positive asymptotically stable realizations (3.8) of the transfer function

$$T(s) = \frac{2s^2 + 12s + 26}{s^2 + 5s + 6}. \tag{3.16}$$

The transfer function (3.16) satisfies the conditions (3.11) and (3.12) since

$a_1^2 - 4a_0 = 1 > 0$ and

$b_2 = 2, \;\; b_0 - a b_1 + (a a_1 - a_0) b_2 = 14 - 2a \geq 0, \;\; b_1 - a_1 b_2 = 2 > 0$ for $0 \leq a \leq 7$.

Using (3.9) we obtain

$$A_{M1} = \begin{bmatrix} -a & 5a - a^2 - 6 \\ 1 & a - 5 \end{bmatrix}, \;\; B_1 = \begin{bmatrix} 14 - 2a \\ 2 \end{bmatrix}, \;\; C_1 = [0 \quad 1], \;\; D_1 = [2] \tag{3.17a}$$

and

$$A_{M2} = \begin{bmatrix} -a & 1 \\ 5a - a^2 - 6 & a - 5 \end{bmatrix}, \;\; B_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \;\; C_2 = [14 - 2a \quad 2], \;\; D_2 = [2] \tag{3.17b}$$

for the parameter a satisfying $2 \leq a \leq 3$. The desired set of positive asymptotically stable realizations of (3.16) is given by

$$\bar{A}_{M1} = P \begin{bmatrix} -a & 5a - a^2 - 6 \\ 1 & a - 5 \end{bmatrix} P^{-1}, \;\; \bar{B}_1 = P \begin{bmatrix} 14 - 2a \\ 2 \end{bmatrix}, \;\; \bar{C}_1 = [0 \quad 1] P^{-1}, \;\; \bar{D}_1 = [2] \tag{3.18a}$$

and

$$\overline{A}_{M2} = P\begin{bmatrix} -a & 1 \\ 5a - a^2 - 6 & a - 5 \end{bmatrix} P^{-1}, \quad \overline{B}_2 = P\begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \overline{C}_2 = [14 - 2a \quad 2]P^{-1}, \quad \overline{D}_2 = [2] \tag{3.18b}$$

where $P \in \Re_+^{2\times 2}$ is any monomial matrix.

Theorem 3.3. Let the transfer function

$$T(s) = \frac{b_3 s^3 + b_2 s^2 + b_1 s + b_0}{s^3 + a_2 s^2 + a_1 s + a_0} \tag{3.19}$$

have only real negative poles $-\alpha_1, -\alpha_2, -\alpha_3$, i.e.

$$d_3(s) = (s + \alpha_1)(s + \alpha_2)(s + \alpha_3) = s^3 + a_2 s^2 + a_1 s + a_0 \tag{3.20a}$$

where

$$a_2 = \alpha_1 + \alpha_2 + \alpha_3, \quad a_1 = \alpha_1(\alpha_2 + \alpha_3) + \alpha_2\alpha_3, \quad a_0 = \alpha_1\alpha_2\alpha_3. \tag{3.20b}$$

There exists the set of positive asymptotically stable realizations

$$\overline{A}_{Mk} = PA_{Mk}P^{-1} \in M_{3s}, \ \overline{B}_k = PB_k \in \Re_+^{3\times 1}, \ \overline{C}_k = C_k P^{-1} \in \Re_+^{1\times 3},$$
$$\overline{D}_k = D_k = [b_3] \in \Re_+^{1\times 1}, \ k = 1,2 \tag{3.21}$$

for any monomial matrix $P \in \Re_+^{3\times 3}$ and A_{Mk}, B_k, C_k, D_k having one of the forms

$$A_{M1} = \begin{bmatrix} -\alpha_1 & 1 & 0 \\ 0 & -\alpha_2 & 1 \\ 0 & 0 & -\alpha_3 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad C_1^T = \begin{bmatrix} b_0 - \alpha_1 b_1 + \alpha_1^2 b_2 + (a_1\alpha_1 - a_0 + a_2\alpha_1^2)b_3 \\ b_1 - (\alpha_1 + \alpha_2)b_2 + [a_2(\alpha_1 + \alpha_2) - a_1]b_3 \\ b_2 - a_2 b_3 \end{bmatrix}, \quad D_1 = [b_3] \tag{3.22a}$$

or

$$A_{M2} = A_{M1}^T, \quad B_2 = C_1^T, \quad C_2 = B_1^T, \quad D_2 = D_1 \tag{3.22b}$$

of the transfer function (3.19) if and only if the conditions

$$b_0 - \alpha_1 b_1 + \alpha_1^2 b_2 + (a_1\alpha_1 - a_0 + a_2\alpha_1^2)b_3 \geq 0 \tag{3.23a}$$

$$b_1 - (\alpha_1 + \alpha_2)b_2 + [a_2(\alpha_1 + \alpha_2) - a_1]b_3 \geq 0 \tag{3.23b}$$

$$b_2 - a_2 b_3 \geq 0 \tag{3.23c}$$

are met.

Proof. The matrix $A_{M1} \in M_{3s}$ if and only if $\alpha_k > 0$ for $k = 1,2,3$. The matrix

$$D_1 = \lim_{s\to\infty} T(s) = [b_3] \in \Re_+^{1\times 1}$$

if and only if $b_3 \geq 0$. The strictly proper transfer function has the form

$$T_{sp}(s) = T(s) - D_1 = \frac{\bar{b}_2 s^2 + \bar{b}_1 s + \bar{b}_0}{s^3 + a_2 s^2 + a_1 s + a_0} \tag{3.24}$$

where $\bar{b}_2 = b_2 - a_2 b_3$, $\bar{b}_1 = b_1 - a_1 b_3$, $\bar{b}_0 = b_0 - a_0 b_3$.

Assuming $B_1^T = [0 \quad 0 \quad 1]$ we obtain

$$T_{sp}(s) = C_1[I_3 s - A_{M1}]^{-1} B_1 = [c_1 \quad c_2 \quad c_3] \begin{bmatrix} s+\alpha_1 & -1 & 0 \\ 0 & s+\alpha_2 & -1 \\ 0 & 0 & s+\alpha_3 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$= \frac{[c_1 \quad c_2 \quad c_3]}{s^3 + a_2 s^2 + a_1 s + a_0} \begin{bmatrix} 1 \\ s+\alpha_1 \\ (s+\alpha_1)(s+\alpha_2) \end{bmatrix} = \frac{c_3 s^2 + [c_2 + c_3(\alpha_1 + \alpha_2)]s + c_1 + \alpha_1 c_2 + \alpha_1 \alpha_2 c_3}{s^3 + a_2 s^2 + a_1 s + a_0} \tag{3.25}$$

From comparison of (3.24) and (3.25) we have

$$\begin{aligned} c_3 &= \bar{b}_2 = b_2 - a_2 b_3, \\ c_2 &= \bar{b}_1 - c_3(\alpha_1 + \alpha_2) = b_1 - a_1 b_3 - c_3(\alpha_1 + \alpha_2) = b_1 - (\alpha_1 + \alpha_2) b_2 + [a_2(\alpha_1 + \alpha_2) - a_1] b_3, \\ c_1 &= \bar{b}_0 = b_0 - a_0 b_3 - c_2 \alpha_1 - c_3 \alpha_1 \alpha_2 = b_0 - \alpha_1 b_1 + \alpha_1^2 b_2 + (a_1 \alpha_1 - a_0 + a_2 \alpha_1^2) b_3. \end{aligned} \tag{3.26}$$

From (3.26) it follows that $C_1 \in \Re_+^{1\times 3}$ if and only if the conditions (3.23) are met. The proof for (3.22b) follows immediately from the equality that

$$\begin{aligned} T(s) = T^T(s) &= [C_1[I_3 s - A_{M1}]^{-1} B_1 + D_1]^T = B_1^T [I_3 s - A_{M1}^T]^{-1} C_1^T + D_1 \\ &= C_2[I_3 s - A_{M2}]^{-1} B_2 + D_2. \end{aligned} \tag{2.27}$$

By Lemma 2.3 the matrices (3.21) are a positive asymptotically stable realization of (3.19) for any monomial matrix $P \in \Re_+^{3\times 3}$ if and only if the matrices (3.22) are its positive asymptotically stable realization.

Theorem 3.4. There exists the set of positive asymptotically stable realizations

$$\bar{A}_{Mk} = P A_{Mk} P^{-1} \in M_{ns}, \ \bar{B}_k = P B_k \in \Re_+^{n\times 1}, \ \bar{C}_k = C_k P^{-1} \in \Re_+^{1\times n}, \ \bar{D}_k = D_k \in \Re_+^{1\times 1}, \quad k = 1,2 \tag{3.28}$$

for any monomial matrix $P \in \Re_+^{n\times n}$ and A_{Mk}, B_k, C_k, D_k having one of the forms

$$A_{M1}=\begin{bmatrix}-\alpha_1 & 1 & 0 & \dots & 0\\ 0 & -\alpha_2 & 1 & \dots & 0\\ \vdots & \vdots & \vdots & \dots & \vdots\\ 0 & 0 & 0 & \dots & -\alpha_n\end{bmatrix},\quad B_1=\begin{bmatrix}0\\ \vdots\\ 0\\ 1\end{bmatrix},\quad C_1^T=\begin{bmatrix}\bar{b}_0-\bar{a}_{10}c_2-\bar{a}_{20}c_3-\dots-\bar{a}_{n-1,0}c_n\\ \vdots\\ \bar{b}_{n-2}-\bar{a}_{n-1,n-2}c_n\\ \bar{b}_{n-1}\end{bmatrix},\quad D_1=[b_n] \tag{3.29a}$$

or

$$A_{M2}=A_{M1}^T,\quad B_2=C_1^T,\quad C_2=B_1^T,\quad D_2=D_1 \tag{3.29b}$$

of the transfer function (3.1) with only real negative poles $-\alpha_1, -\alpha_2, \dots, -\alpha_n$ if and only if the conditions

$$\begin{aligned}&c_n=b_{n-1}-a_{n-1}b_n\ge 0\\ &c_{n-1}=b_{n-2}-a_{n-2}b_n-\bar{a}_{n-1,n-2}c_n\ge 0\\ &\vdots\\ &c_1=b_0-a_0b_n-\bar{a}_{10}c_2-\bar{a}_{20}c_3-\dots-\bar{a}_{n-1,0}c_n\ge 0\end{aligned} \tag{3.30a}$$

where

$$\begin{aligned}&\bar{a}_{10}=\alpha_1,\ \bar{a}_{20}=\alpha_1\alpha_2,\ \bar{a}_{21}=\alpha_1+\alpha_2,\ \bar{a}_{30}=\alpha_1\alpha_2\alpha_3,\ \bar{a}_{31}=\alpha_1(\alpha_2+\alpha_3)+\alpha_2\alpha_3,\ \bar{a}_{32}=\alpha_1+\alpha_2+\alpha_3,\\ &\vdots\\ &\bar{a}_{n-1,0}=\alpha_1\alpha_2\dots\alpha_n,\ \bar{a}_{n-1,1}=\alpha_1(\alpha_2+\alpha_3+\dots+\alpha_n)+\alpha_2(\alpha_3+\alpha_4+\dots+\alpha_n)+\dots+\alpha_{n-1}\alpha_n,\dots,\\ &\bar{a}_{n-1,n-2}=\alpha_1+\alpha_2+\dots+\alpha_n\end{aligned} \tag{3.30b}$$

are met.

Proof. The matrix $A_{M1}\in M_{ns}$ if and only $\alpha_k>0$ for $k=1,2,\dots,n$. The matrix

$$D_1=\lim_{s\to\infty}T(s)=[b_n]\in\Re_+^{1\times 1} \tag{3.31}$$

if and only if $b_n\ge 0$. The strictly proper transfer function has the form

$$T_{sp}(s)=T(s)-D_1=\frac{\bar{b}_{n-1}s^{n-1}+\dots+\bar{b}_1s+\bar{b}_0}{s^n+a_{n-1}s^{n-1}+\dots+a_1s+a_0} \tag{3.32a}$$

where

$$\bar{b}_k=b_k-a_kb_n \text{ for } k=0,1,\dots,n-1. \tag{3.32b}$$

Assuming $B_1^T=[0\quad\dots\quad 0\quad 1]\in\Re_+^{n\times 1}$ we obtain

$$T_{sp}(s) = C_1[I_n s - A_{M1}]^{-1} B_1 = [c_1 \quad \dots \quad c_n] \begin{bmatrix} s+\alpha_1 & -1 & 0 & \dots & 0 \\ 0 & s+\alpha_2 & -1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & s+\alpha_n \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

$$= \frac{[c_1 \quad \dots \quad c_n]}{d_n(s)} \begin{bmatrix} 1 \\ p_1(s) \\ \vdots \\ p_{n-1}(s) \end{bmatrix} = \frac{c_1 + c_2 p_1(s) + \dots + c_n p_{n-1}(s)}{d_n(s)} \tag{3.33a}$$

where

$$\begin{aligned} &d_n(s) = s^n + a_{n-1}s^{n-1} + \dots + a_1 s + a_0, \\ &p_1(s) = s + \alpha_1 = s + \overline{a}_{10}, \quad \overline{a}_{10} = \alpha_1, \\ &p_2(s) = (s+\alpha_1)(s+\alpha_2) = s^2 + \overline{a}_{21}s + \overline{a}_{20}, \quad \overline{a}_{21} = \alpha_1 + \alpha_2, \quad \overline{a}_{20} = \alpha_1\alpha_2, \\ &\vdots \\ &p_{n-1}(s) = (s+\alpha_1)(s+\alpha_2)\dots(s+\alpha_{n-1}) = s^{n-1} + \overline{a}_{n-1,n-2}s^{n-2} + \dots + \overline{a}_{n-1,1}s + \overline{a}_{n-1,0}, \\ &\quad \overline{a}_{n-1,n-2} = \alpha_1 + \alpha_2 + \dots + \alpha_{n-1}, \dots, \quad \overline{a}_{n-1,0} = \alpha_1\alpha_2 \dots, \alpha_{n-1}. \end{aligned} \tag{3.33b}$$

From comparison of (3.33a) and (3.32a) we have

$$\begin{aligned} &c_n = \overline{b}_{n-1} = b_{n-1} - a_{n-1}b_n, \\ &c_{n-1} = \overline{b}_{n-2} - \overline{a}_{n-1,n-2}c_n = b_{n-2} - a_{n-2}b_n - \overline{a}_{n-1,n-2}c_n, \\ &\vdots \\ &c_1 = \overline{b}_0 - \overline{a}_{10}c_2 - \overline{a}_{20}c_3 - \dots - \overline{a}_{n-1,0}c_n. \end{aligned} \tag{3.34}$$

From (3.34) it follows that $C_1 \in \Re_+^{1\times n}$ if and only if the conditions (3.30) are met. The proof for (3.29b) follows immediately from (2.27). By Lemma 2.3 the matrices (3.28) are a positive asymptotically stable realization of (3.1) for any monomial matrix $P \in \Re_+^{n\times n}$ if and only if the matrices (3.29) are its positive asymptotically stable realization.

From above considerations we have the following procedure for computation of the set of positive asymptotically stable realizations (3.28) of the transfer function (3.1) with real negative poles.

Procedure 3.1.

Step 1. Check the conditions (3.30). If the conditions are met, go to Step 2, if not then does not exist the set of realizations.

Step 2. Using (3.29) compute the matrices A_{Mk}, B_k, C_k, D_k for example for $k = 1$ or $k = 2$.

Step 3. Using (3.28) compute the desired set of realizations.

Example 3.2. Compute the set of positive asymptotically table realizations of the transfer function

$$T(s) = \frac{0.2s^4 + 2.2s^3 + 8.6s^2 + 12.4s + 7.8}{s^4 + 6s^3 + 13s^2 + 12s + 4}. \tag{3.35}$$

The transfer function (3.35) has two real double poles $-\alpha_1 = -\alpha_2 = -1$, $-\alpha_3 = -\alpha_4 = -2$. Using Procedure 3.1 we obtain the following.

Step 1. The conditions (3.30) are satisfied since

$$\begin{aligned} c_4 &= b_3 - a_3 b_4 = 1 > 0, \\ c_3 &= b_2 - a_2 b_4 - \bar{a}_{32} c_4 = 2 > 0, \\ c_2 &= b_1 - a_1 b_4 - \bar{a}_{21} c_3 - \bar{a}_{31} c_4 = 1 > 0, \\ c_1 &= b_0 - a_0 b_4 - \bar{a}_{10} c_2 - \bar{a}_{20} c_3 - \bar{a}_{30} c_4 = 2 > 0. \end{aligned} \tag{3.36}$$

Step 2. In this case the matrices (3.29a) have the forms

$$A_{M1} = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -2 & 1 \\ 0 & 0 & 0 & -2 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad C_1 = [2 \quad 1 \quad 2 \quad 1], \quad D_1 = [0.2]. \tag{3.37}$$

Step 3. The desired set of realizations of (3.35) is given by

$$\bar{A}_{M1} = P \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -2 & 1 \\ 0 & 0 & 0 & -2 \end{bmatrix} P^{-1}, \quad \bar{B}_1 = P \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad \bar{C}_1 = [2 \quad 1 \quad 2 \quad 1] P^{-1}, \quad \bar{D}_1 = [0.2] \tag{3.38}$$

for any monomial matrix $P \in \Re_+^{4\times 4}$.

Systems with complex conjugate poles

In this section the single-input single-output linear continuous-time system with the transfer function (3.1) having at least one pair of complex conjugate poles will be considered.

Theorem 4.1. There exists the set of positive asymptotically stable realizations

$$\overline{A}_{Mk} = PA_{Mk}P^{-1} \in M_{3s},\ \overline{B}_k = PB_k \in \Re_+^{3\times 1},\ \overline{C}_k = C_kP^{-1} \in \Re_+^{1\times 3},$$
$$\overline{D}_k = D_k = [b_3] \in \Re_+^{1\times 1},\ k = 1,2 \qquad (4.1)$$

for any monomial matrix $P \in \Re_+^{3\times 3}$ and the matrices A_{Mk}, B_k, C_k, D_k having one of the forms

$$A_{M1} = \begin{bmatrix} p_1 + p_2 - a_2 & 1 & a_{13} \\ 0 & -p_1 & a_{23} \\ 1 & 0 & -p_2 \end{bmatrix},\ B_1 = \begin{bmatrix} b_1 + (p_2 - a_2)b_2 + (a_2^2 - a_1 - a_2p_2)b_3 \\ b_0 - p_1b_1 + p_1^2b_2 + (a_1p_1 - a_0 - a_2p_1^2)b_3 \\ b_2 - a_2b_3 \end{bmatrix},$$
$$C_1 = [0 \quad 0 \quad 1],\ D_1 = [b_3],$$
$$a_{13} = (a_2 - p_1 - p_2)(p_1 + p_2) + p_1p_2 - a_1,\ a_{23} = (a_2 - p_1 - p_2)p_1p_2 - a_{13}p_1 - a_0 \qquad (4.2a)$$

or

$$A_{M2} = A_{M1}^T,\ B_2 = C_1^T,\ C_2 = B_1^T,\ D_2 = D_1 \qquad (4.2b)$$

of the transfer function

$$T(s) = \frac{b_3s^3 + b_2s^2 + b_1s + b_0}{s^3 + a_2s^2 + a_1s + a_0} \qquad (4.3)$$

if and only if the coefficients of the polynomial

$$d_3(s) = s^3 + a_2s^2 + a_1s + a_0 \qquad (4.4)$$

satisfies the conditions

$$a_2^2 - 3a_1 \geq 0,\ -2a_2^3 + 9a_1a_2 - 27a_0 \geq 0 \qquad (4.5)$$

and

$$\begin{aligned} &b_1 + (p_2 - a_2)b_2 + (a_2^2 - a_1 - a_2p_2)b_3 \geq 0, \\ &b_0 - p_1b_1 + p_1^2b_2 + (a_1p_1 - a_0 - a_2p_1^2)b_3 \geq 0, \\ &b_2 - a_2b_3 \geq 0 \end{aligned} \qquad (4.6)$$

are where p_1, p_2 are positive parameters satisfying $0 < p_1 + p_2 < a_2$.

Proof. If the matrix $A_{M1} \in M_{3s}$ if and only if its characteristic polynomial

$$d_3(s) = \det[I_3 s - A_{M1}] = \begin{vmatrix} s + a_2 - p_1 - p_2 & -1 & -a_{13} \\ 0 & s + p_1 & -a_{23} \\ -1 & 0 & s + p_2 \end{vmatrix} = s^3 + a_2 s^2 + a_1 s + a_0 \tag{4.7}$$

has the coefficients satisfying the conditions (4.5) [19] and $0 < p_1 + p_2 < a_2$.

The matrix

$$D_1 = \lim_{s \to \infty} T(s) = [b_3] \in \Re_+^{1 \times 1} \tag{4.8}$$

if and only if $b_3 \geq 0$. The strictly proper transfer function has the form

$$T_{sp}(s) = T(s) - D_1 = \frac{\bar{b}_2 s^2 + \bar{b}_1 s + \bar{b}_0}{s^3 + a_2 s^2 + a_1 s + a_0} \tag{4.9a}$$

where

$$\bar{b}_2 = b_2 - a_2 b_3, \ \bar{b}_1 = b_1 - a_1 b_3, \ \bar{b}_0 = b_0 - a_0 b_3. \tag{4.9b}$$

Assuming $C_1 = [0 \quad 0 \quad 1]$ we obtain

$$\begin{aligned} T_{sp}(s) = C_1[I_3 s - A_{M1}]^{-1} B_1 &= [0 \quad 0 \quad 1] \begin{bmatrix} s + a_2 - p_1 - p_2 & -1 & -a_{13} \\ 0 & s + p_1 & -a_{23} \\ -1 & 0 & s + p_2 \end{bmatrix}^{-1} \begin{bmatrix} b_{11} \\ b_{12} \\ b_{13} \end{bmatrix} \\ &= \frac{[s + p_1 \quad 1 \quad (s + a_2 - p_1 - p_2)(s + p_1)]}{s^3 + a_2 s^2 + a_1 s + a_0} \begin{bmatrix} b_{11} \\ b_{12} \\ b_{13} \end{bmatrix} \\ &= \frac{b_{13} s^2 + [b_{11} + (a_2 - p_2) b_{13}] s + b_{12} + p_1 b_{11} + (a_2 - p_1 - p_2) p_1 b_{13}}{s^3 + a_2 s^2 + a_1 s + a_0}. \end{aligned} \tag{4.10}$$

From comparison of (4.9a) and (4.10) we have

$$\begin{aligned} b_{13} &= \bar{b}_2 = b_2 - a_2 b_3, \\ b_{11} &= \bar{b}_1 - (a_2 - p_2) b_{13} = b_1 + (p_2 - a_2) b_2 + (a_2^2 - a_1 - a_2 p_2) b_3, \\ b_{12} &= \bar{b}_0 - p_1 b_{11} - (a_2 - p_1 - p_2) p_1 b_{13} = b_0 - p_1 b_1 + p_1^2 b_2 + (a_1 p_1 - a_0 - a_2 p_1^2) b_3. \end{aligned} \tag{4.11}$$

From (4.11) it follows that $B_1 \in \Re_+^{3 \times 1}$ if and only if the conditions (4.6) are met. The proof for (4.2b) is similar. By Lemma 2.3 the matrices (4.1) are a positive asymptotically stable

realization for any monomial matrix $P \in \Re_+^{3\times 3}$ of (4.3) if and only if the matrices (4.2) are its positive asymptotically stable realization.

Remark 4.1. The matrix A_{M1} in Theorem 4.1 can be replaced by the matrices [19]

$$A_{M3} = \begin{bmatrix} p_1 + p_2 - a_2 & 0 & 1 \\ a_{21} & -p_1 & 0 \\ a_{31} & 1 & -p_2 \end{bmatrix}, \quad A_{M4} = \begin{bmatrix} p_1 + p_2 - a_2 & a_{12} & 0 \\ 0 & -p_1 & 1 \\ 1 & a_{32} & -p_2 \end{bmatrix} \tag{4.12a}$$

and the matrix A_{M2} by A_{M3}^T, A_{M4}^T. For A_{M3} the matrices B_3 and C_3 have the forms

$$B_3 = \begin{bmatrix} b_2 - a_2 b_3 \\ b_0 - p_1 b_1 + p_1^2 b_2 + (a_1 p_1 - a_0 - a_2 p_1^2) b_3 \\ b_1 - (p_1 + p_2) b_2 + [a_2 (p_1 + p_2) - a_1] b_3 \end{bmatrix}, \quad C_3^T = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{4.12b}$$

and for A_{M4} the matrices B_4 and C_4 have the forms

$$B_4 = \begin{bmatrix} b_0 + (p_1 + p_2 - a_2) b_1 + [(p_1 + p_2)^2 - a_2^2] b_2 + (p_1 + p_2 - a_2)[a_2^2 - a_1 - a_2 (p_1 + p_2)] b_3 \\ b_2 - a_2 b_3 \\ b_1 + (p_2 + a_2) b_2 + (a_2^2 - a_1 - a_2 p_2) b_3 \end{bmatrix}, \quad C_4^T = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tag{4.12c}$$

From above considerations we have the following procedure for computation of the set of positive asymptotically stable realizations.

Procedure 4.1.

Step 1. Check the conditions (4.5) and (4.6). If the conditions are met, go to Step 2, if not then does not exist the set of realizations.

Step 2. Using (4.2) compute the matrices A_{Mk}, B_k, C_k, D_k for example for $k = 1$ or $k = 2$.

Step 3. Using (4.1) compute the desired set of realizations.

Example 4.1. Compute the set of positive asymptotically table realizations of the transfer function

$$T(s) = \frac{0.1s^3 + s^2 + 4s + 12}{s^3 + 9s^2 + 25s + 17}. \tag{4.13}$$

Using Procedure 4.1 we obtain the following.

Step 1. The transfer function (4.13) satisfies the conditions (4.5) and (4.6) since

$$\begin{aligned} &a_2^2 - 3a_1 = 6 > 0, \\ &-2a_2^3 + 9a_1 a_2 - 27a_0 = 108 > 0 \end{aligned} \tag{4.14a}$$

and

$$
\begin{aligned}
&b_1 + (p_2 - a_2)b_2 + (a_2^2 - a_1 - a_2 p_2)b_3 = 0.6 + 0.1p_2 > 0,\\
&b_0 - p_1 b_1 + p_1^2 b_2 + (a_1 p_1 - a_0 - a_2 p_1^2)b_3 = 10.3 + p_1(0.1p_1 - 1.5) > 0,\\
&b_2 - a_2 b_3 = 0.1 > 0
\end{aligned}
\tag{4.14b}
$$

for $0 < p_1 + p_2 < 9$.

Step 2. Using (4.2a), (4.13) and (4.14b) we obtain

$$
A_{M1} = \begin{bmatrix} p_1 + p_2 - 9 & 1 & a_{13} \\ 0 & -p_1 & a_{23} \\ 1 & 0 & -p_2 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0.6 + 0.1p_2 \\ 10.3 + p_1(0.1p_1 - 1.5) \\ 0.1 \end{bmatrix}, \quad C_1^T = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad D_1 = [0.1]
\tag{4.15}
$$

where

$$
\begin{aligned}
a_{13} &= [9 - (p_1 + p_2)](p_1 + p_2) + p_1 p_2 - 25,\\
a_{23} &= [8 - (p_1 + p_2)]p_1 p_2 - 9(p_1 + p_2) + (p_1 + p_2)^2 + 8
\end{aligned}
$$

and p_1, p_2 are arbitrary parameters satisfying $0 < p_1 + p_2 < 9$.

Step 3. The desired set of positive stable realizations is given by

$$
\overline{A}_{M1} = PA_{M1}P^{-1}, \quad \overline{B}_1 = PB_1, \quad \overline{C}_1 = C_1 P^{-1}, \quad \overline{D}_1 = D_1
\tag{4.16}
$$

for any monomial matrix $P \in \Re_+^{3\times 3}$.

Theorem 4.2. There exists the set of positive asymptotically stable realizations

$$
\overline{A}_{Mk} = PA_{Mk}P^{-1} \in M_{4s}, \ \overline{B}_k = PB_k \in \Re_+^{4\times 1}, \ \overline{C}_k = C_k P^{-1} \in \Re_+^{1\times 4}, \ \overline{D}_k = D_k \in \Re_+^{1\times 1}
\tag{4.17}
$$

for any monomial matrix $P \in \Re_+^{4\times 4}$ and the matrices A_{Mk}, B_k, C_k, D_k having one of the forms

$$
A_{M1} = \begin{bmatrix} -p_1 & 1 & 0 & a_{14} \\ 0 & -p_2 & 1 & a_{24} \\ 0 & 0 & -p_3 & a_{34} \\ 1 & 0 & 0 & p_1 + p_2 + p_3 - a_3 \end{bmatrix}, \quad B_1 = \begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \\ \tilde{b}_4 \end{bmatrix}, \quad C_1 = [0 \quad 0 \quad 0 \quad 1], \quad D_1 = [b_4],
$$

$$
\begin{aligned}
\tilde{b}_1 &= b_2 - (p_1 + p_2 + p_3)b_3 + [a_3(p_1 + p_2 + p_3) - a_2]b_4\\
\tilde{b}_2 &= b_1 - (p_2 + p_3)b_2 + (p_2^2 + p_3^2 + p_2 p_3)b_3 + [a_3(p_2 + p_3) - a_1 - a_3(p_2^2 + p_3^2 + p_2 p_3)]b_4\\
\tilde{b}_3 &= b_0 - p_3 b_1 + p_3^2 b_2 - p_3^2 b_3 + [a_1 p_3 - a_0 + a_2 p_2 p_3 - a_3 p_2 p_3]b_4\\
\tilde{b}_4 &= b_3 - a_3 b_4
\end{aligned}
\tag{4.18a}
$$

and

$$a_{14} = p_1(a_3 - p_1) + p_2(a_3 - p_1 - p_2) + p_3(a_3 - p_1 - p_2 - p_3) - a_2 \geq 0,$$
$$a_{24} = (p_1 + p_2)p_3(a_3 - p_1 - p_2 - p_3) + p_1p_2(a_3 - p_1 - p_2) - a_{14}(p_2 + p_3) - a_1 \geq 0,$$
$$a_{34} = p_1p_2p_3(a_3 - p_1 - p_2 - p_3) - a_{14}p_2p_3 - a_{24}p_3 - a_0 \geq 0$$

(4.18b)

or

$$A_{M2} = A_{M1}^T, \quad B_2 = C_1^T, \quad C_2 = B_1^T, \quad D_2 = D_1 \tag{4.18c}$$

of the transfer function

$$T(s) = \frac{b_4s^4 + b_3s^3 + b_2s^2 + b_1s + b_0}{s^4 + a_3s^3 + a_2s^2 + a_1s + a_0} \tag{4.19}$$

if and only if the coefficients of the polynomial

$$d_4(s) = s^4 + a_3s^3 + a_2s^2 + a_1s + a_0 \tag{4.20}$$

satisfies the conditions

$$3a_3^2 - 8a_2 \geq 0, \quad -a_3^3 + 4a_2a_3 - 8a_1 \geq 0, \quad 3a_3^4 - 16a_2a_3^2 + 64a_1a_3 - 256a_0 \geq 0 \tag{4.21}$$

and

$$b_2 - (p_1 + p_2 + p_3)b_3 + [a_3(p_1 + p_2 + p_3) - a_2]b_4 \geq 0$$
$$b_1 - (p_2 + p_3)b_2 + (p_2^2 + p_3^2 + p_2p_3)b_3 + [a_3(p_2 + p_3) - a_1 - a_3(p_2^2 + p_3^2 + p_2p_3)]b_4 \geq 0$$
$$b_0 - p_3b_1 + p_3^2b_2 - p_3^2b_3 + [a_1p_3 - a_0 + a_2p_2p_3 - a_3p_2p_3]b_4 \geq 0$$
$$b_3 - a_3b_4 \geq 0$$

(4.22)

are where p_1, p_2, p_3 are positive parameters satisfying $0 < p_1 + p_2 + p_3 < a_3$.

The proof is similar to the proof of Theorem 4.1.

Remark 4.2. The matrix A_{M1} in Theorem 4.2 can be replaced by the matrices

$$A_{M3} = \begin{bmatrix} -p_1 & 1 & a_{13} & 0 \\ 0 & -p_2 & a_{23} & 1 \\ 1 & 0 & -p_3 & 0 \\ 0 & 0 & a_{43} & p_1 + p_2 + p_3 - a_3 \end{bmatrix}, \quad A_{M4} = \begin{bmatrix} -p_1 & a_{12} & 1 & 0 \\ 1 & -p_2 & 0 & 0 \\ 0 & a_{32} & -p_3 & 1 \\ 0 & a_{42} & 0 & p_1 + p_2 + p_3 - a_3 \end{bmatrix},$$

$$A_{M5} = \begin{bmatrix} -p_1 & 0 & 0 & 1 \\ a_{21} & -p_2 & 0 & 0 \\ a_{31} & 1 & -p_3 & 0 \\ a_{41} & 0 & 1 & p_1 + p_2 + p_3 - a_3 \end{bmatrix}, \quad 0 < p_1 + p_2 + p_3 < a_3$$

(4.23)

and the matrix A_{M2} by the matrices A_{M3}^T, A_{M4}^T, A_{M5}^T.

In general case let us consider the transfer function

$$T(s) = \frac{b_n s^n + b_{n-1} s^{n-1} + \ldots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_0} \tag{4.24}$$

with at least one pair of complex conjugate poles.

Theorem 4.3. There exists the set of positive asymptotically stable realizations

$$\overline{A}_{Mk} = P A_{Mk} P^{-1} \in M_{ns},\ \overline{B}_k = P B_k \in \Re_+^{n\times 1},\ \overline{C}_k = C_k P^{-1} \in \Re_+^{1\times n},\ \overline{D}_k = D_k \in \Re_+^{1\times 1} \tag{4.25}$$

for any monomial matrix $P \in \Re_+^{n\times n}$ and A_{Mk}, B_k, C_k, D_k having one of the forms

$$A_{M1} = \begin{bmatrix} -p_1 & 1 & 0 & \ldots & 0 & a_{1,n} \\ 0 & -p_2 & 1 & \ldots & 0 & a_{2,n} \\ 0 & 0 & -p_3 & \ldots & 0 & a_{3,n} \\ \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & 1 & a_{n-2,n} \\ 0 & 0 & 0 & \ldots & -p_{n-1} & a_{n-1,n} \\ 1 & 0 & 0 & \ldots & 0 & p_1 + \ldots + p_{n-1} - a_{n-1} \end{bmatrix},$$

$$B_1 = \begin{bmatrix} b_{n-2} - a_{n-2} b_n - \hat{a}_{n,n-2} b_{1,n} \\ \vdots \\ b_0 - a_0 b_n - \hat{a}_{n,0} b_{1,n} - \hat{a}_{10} b_{1,1} - \ldots - \hat{a}_{n-2,0} b_{1,n-2} \\ b_{n-1} - a_{n-1} b_n \end{bmatrix},\ C_1^T = [0 \quad \ldots \quad 0 \quad 1],\ D_1 = [b_n] \tag{4.26a}$$

where $p_1, p_2, \ldots, p_{n-1}$ are positive parameters satisfying $0 < p_1 + p_2 + \ldots + p_{n-1} < a_{n-1}$ or

$$A_{M2} = A_{M1}^T,\ B_2 = C_1^T,\ C_2 = B_1^T,\ D_2 = D_1 \tag{4.26b}$$

and

$$\begin{aligned} a_{1,n} &= p_1(a_{n-1} - p_1) + p_2(a_{n-1} - p_1 - p_2) + \ldots + p_{n-1}(a_{n-1} - p_1 - \ldots - p_{n-1}) - a_{n-2} \\ &\vdots \\ a_{n-1,n} &= p_1 \ldots p_{n-1}(a_{n-1} - p_1 - \ldots - p_{n-1}) - \hat{a}_{10} a_{1,n} - \ldots - \hat{a}_{n-2,0} a_{n-2,n} \end{aligned} \tag{4.26c}$$

of the transfer function (4.24) if and only if the coefficients of the polynomial

$$d_n(s) = s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_0 \tag{4.27}$$

satisfies the conditions

$$
\begin{aligned}
&C_2^n\left(\frac{a_{n-1}}{n}\right)^2 - a_{n-2} \geq 0,\\
&C_3^n\left(\frac{a_{n-1}}{n}\right)^3 - \left[C_2^n\left(\frac{a_{n-1}}{n}\right)^2 - a_{n-2}\right]C_1^{n-2}\left(\frac{a_{n-1}}{n}\right) - a_{n-3} \geq 0,\\
&\vdots\\
&C_n^n\left(\frac{a_{n-1}}{n}\right)^n - \left[C_2^n\left(\frac{a_{n-1}}{n}\right)^2 - a_{n-2}\right]C_1^{n-2}\left(\frac{a_{n-1}}{n}\right)^{n-2} - \ldots - C_1^1\left(\frac{a_{n-1}}{n}\right) - a_0 \geq 0\\
&C_k^n = \frac{n!}{k!(n-k)!}
\end{aligned}
\tag{4.28}
$$

and

$$
\begin{aligned}
&b_{n-2} - a_{n-2}b_n - \hat{a}_{n,n-2}b_{1,n} \geq 0\\
&\vdots\\
&b_0 - a_0 b_n - \hat{a}_{n,0}b_{1,n} - \hat{a}_{10}b_{11} - \ldots - \hat{a}_{n-2,0}b_{1,n-2} \geq 0\\
&b_{n-1} - a_{n-1}b_n \geq 0
\end{aligned}
\tag{4.29a}
$$

where

$$
\begin{aligned}
&\hat{a}_{10} = p_2 p_3 \ldots p_{n-1}, \quad \hat{a}_{20} = p_3 p_4 \ldots p_{n-1}, \ldots, \hat{a}_{n,0} = p_1 p_2 \ldots p_{n-1},\\
&\vdots\\
&\hat{a}_{1,n-3} = p_2 + p_3 + \ldots + p_{n-1}, \quad \hat{a}_{2,n-4} = p_3 + p_4 + \ldots + p_{n-1}, \ldots, \hat{a}_{n,n-2} = p_1 + p_2 + \ldots + p_{n-1}.
\end{aligned}
\tag{4.29b}
$$

Proof. It is well-known [Kaczorek 2012] that there exists $A_{M1} \in M_{ns}$ if and only if the coefficients of the polynomial (4.27) are positive and satisfy the conditions (4.28). The matrix

$$D_1 = \lim_{s \to \infty} T(s) = [b_n] \in \Re_+^{1 \times 1} \tag{4.30}$$

if and only if $b_n \geq 0$. The strictly proper transfer function has the form

$$T_{sp}(s) = T(s) - D_1 = \frac{\bar{b}_{n-1}s^{n-1} + \ldots + \bar{b}_1 s + \bar{b}_0}{s^n + a_{n-1}s^{n-1} + \ldots + a_1 s + a_0} \tag{4.31a}$$

where

$$\bar{b}_k = b_k - a_k b_n \text{ for } k = 1,2,\ldots,n-1. \tag{4.31b}$$

Assuming $C_1 = [0 \quad \ldots \quad 0 \quad 1] \in \Re_+^{1\times n}$ we obtain

$$\begin{aligned} T_{sp}(s) &= C_1[I_n s - A_{M1}]^{-1} B_1 \\ &= [0 \quad \ldots \quad 0 \quad 1] \begin{bmatrix} s+p_1 & -1 & 0 & \ldots & 0 & -a_{1,n} \\ 0 & s+p_2 & -1 & \ldots & 0 & -a_{2,n} \\ 0 & 0 & s+p_3 & \ldots & 0 & -a_{3,n} \\ \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & -1 & -a_{n-2,n} \\ 0 & 0 & 0 & \ldots & s+p_{n-1} & -a_{n-1,n} \\ -1 & 0 & 0 & \ldots & 0 & s+a_{n-1}-p_1-\ldots-p_{n-1} \end{bmatrix}^{-1} \begin{bmatrix} b_{11} \\ \vdots \\ b_{1,n-1} \\ b_{1,n} \end{bmatrix} \\ &= \frac{[p_1(s) \quad \ldots \quad p_n(s)]}{d_n(s)} \begin{bmatrix} b_{11} \\ \vdots \\ b_{1,n-1} \\ b_{1,n} \end{bmatrix} = \frac{p_1(s)b_{11} + p_2(s)b_{12} + \ldots + p_n(s)b_{1,n}}{d_n(s)} \end{aligned} \tag{4.32a}$$

where

$$\begin{aligned} &p_1(s) = (s+p_2)(s+p_3)\ldots(s+p_{n-1}) = s^{n-2} + \hat{a}_{1,n-3}s^{n-3} + \ldots + \hat{a}_{11}s + \hat{a}_{10}, \\ &\quad \hat{a}_{1,n-3} = p_2 + p_3 + \ldots + p_{n-1}, \ldots, \hat{a}_{10} = p_2 p_3 \ldots p_{n-1}, \\ &p_2(s) = (s+p_3)(s+p_4)\ldots(s+p_{n-1}) = s^{n-3} + \hat{a}_{2,n-4}s^{n-4} + \ldots + \hat{a}_{21}s + \hat{a}_{20}, \\ &\quad \hat{a}_{2,n-4} = p_3 + p_4 + \ldots + p_{n-1}, \ldots, \hat{a}_{20} = p_3 p_4 \ldots p_{n-1}, \\ &\vdots \\ &p_{n-2}(s) = s + p_{n-1} = s + \hat{a}_{n-2,0}, \quad \hat{a}_{n-2,0} = p_{n-1}, \\ &p_{n-1}(s) = 1 \\ &d_n(s) = (s+p_1)(s+p_2)\ldots(s+p_{n-1}) = s^{n-1} + \hat{a}_{n,n-2}s^{n-2} + \ldots + \hat{a}_{n,1}s + \hat{a}_{n,0}, \\ &\quad \hat{a}_{n,n-2} = p_1 + p_2 + \ldots + p_{n-1}, \ldots, \hat{a}_{n,0} = p_1 p_2 \ldots p_{n-1}. \end{aligned} \tag{4.32b}$$

From comparison of (4.31a) and (4.32a) we have

$$\begin{aligned} &b_{1,n} = \bar{b}_{n-1} = b_{n-1} - a_{n-1}b_n, \\ &b_{11} = \bar{b}_{n-2} - \hat{a}_{n,n-2}b_{1,n} = b_{n-2} - a_{n-2}b_n - \hat{a}_{n,n-2}b_{1,n}, \\ &\vdots \\ &b_{1,n-1} = \bar{b}_0 - \hat{a}_{n,0}b_{1,n} - \hat{a}_{10}b_{11} - \ldots - \hat{a}_{n-2,0}b_{1,n-2} = b_0 - a_0 b_n - \hat{a}_{n,0}b_{1,n} - \hat{a}_{10}b_{11} - \ldots - \hat{a}_{n-2,0}b_{1,n-2}. \end{aligned} \tag{4.33}$$

From (4.33) it follows that $B_1 \in \Re_+^{n\times 1}$ if and only if the conditions (4.29a) are met. The proof for (4.26b) follows immediately from (4.27). By Lemma 2.3 the matrices (4.25) are a positive asymptotically stable realization of (4.24) for any monomial matrix $P \in \Re_+^{n\times n}$ if and only if the matrices (4.26) are its positive asymptotically stable realization.

Remark 4.2. The matrix A_{M1} in Theorem 4.2 can be replaced by the matrices

$$A_{M3} = \begin{bmatrix} -p_1 & 1 & 0 & \dots & a_{1,n-1} & 0 \\ 0 & -p_2 & 1 & \dots & a_{2,n-1} & 1 \\ 0 & 0 & -p_3 & \dots & a_{3,n-1} & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & a_{n-2,n-1} & 0 \\ 1 & 0 & 0 & \dots & -p_{n-1} & 0 \\ 0 & 0 & 0 & \dots & a_{n,n-1} & p_1+\dots+p_{n-1}-a_{n-1} \end{bmatrix},\dots,$$

$$A_{Mn+1} = \begin{bmatrix} -p_1 & 0 & 0 & \dots & 0 & 1 \\ a_{21} & -p_2 & 0 & \dots & 0 & 0 \\ a_{31} & 1 & -p_3 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ a_{n-2,1} & 0 & 0 & \dots & 0 & 0 \\ a_{n-1,1} & 0 & 0 & \dots & -p_{n-1} & 0 \\ a_{n,1} & 0 & 0 & \dots & 1 & p_1+\dots+p_{n-1}-a_{n-1} \end{bmatrix} \qquad (4.23)$$

where $p_1, p_2, \dots, p_{n-1}$ are positive parameters satisfying $0 < p_1 + p_2 + \dots + p_{n-1} < a_{n-1}$ and the matrix A_{M2} by the matrices $A_{M3}^T,\dots,A_{Mn+2}^T$.

To compute the desired set of positive asymptotically stable realizations (4.25) of (4.24) Procedure 4.1 with slight modifications can be used.

Conclusion

The problem of existence and computation of the set of positive asymptotically stable realizations of a proper transfer function of linear continuous-time systems has been formulated and solved. Necessary and sufficient conditions for existence of the set of realizations have been established (Theorems 3.1 – 3.4 and 4.1 – 4.3). Procedure for computation of the set of realizations for transfer functions with only real negative poles and with at least one pair of complex conjugate poles have been proposed

(Procedures 3.1 and 4.1). The effectiveness of the procedures have been demonstrated on numerical examples. The presented methods can be extended to positive asymptotically stable discrete-time linear systems and also to multi-input multi-output continuous-time and discrete-time linear systems. An open problem is an existence of these considerations to fractional linear systems [Kaczorek 2011c].

Acknowledgment

This work was supported under work S/WE/1/11.

Bibliography

[Farina and Rinaldi, 2000] L. Farina, S. Rinaldi. *Positive Linear Systems*, Theory and Applications, J. Wiley, New York, 2000.

[Benvenuti and Farina, 2004] L. Benvenuti, L. Farina. *A tutorial on the positive realization problem*, IEEE Trans. Autom. Control, vol. 49, no. 5, 2004, 651-664.

[Kaczorek, 1992] T. Kaczorek. *Linear Control Systems*, vol.1, Research Studies Press, J. Wiley, New York 1992.

[Kaczorek, 2002] T. Kaczorek. *Positive 1D and 2D Systems*, Springer-Verlag, London, 2002.

[Kaczorek, 2004] T. Kaczorek. *Realization problem for positive discrete-time systems with delay*, System Science, vol. 30, no. 4, 2004, 117-130.

[Kaczorek, 2005] T. Kaczorek. Positive minimal realizations for singular discrete-time systems with delays in state and delays in control, Bull. Pol. Acad. Sci. Techn., vol 53, no. 3, 2005, 293-298.

[Kaczorek, 2006a] T. Kaczorek. *A realization problem for positive continuous-time linear systems with reduced numbers of delays*, Int. J. Appl. Math. Comp. Sci. 2006, Vol. 16, No. 3, pp. 325-331.

[Kaczorek, 2006b] T. Kaczorek. *Computation of realizations of discrete-time cone systems*. Bull. Pol. Acad. Sci. Techn. vol. 54, no. 3, 2006, 347-350.

[Kaczorek, 2006c] T. Kaczorek. Realization problem for positive multivariable discrete-time linear systems with delays in the state vector and inputs, Int. J. Appl. Math. Comp. Sci., vol. 16, no. 2, 2006, 101-106.

[Kaczorek, 2008a] T. Kaczorek. *Fractional positive continuous-time linear systems and their reachability*, Int. J. Appl. Math. Comput. Sci., vol. 18, no. 2, 2008, 223-228.

[Kaczorek, 2008b] T. Kaczorek. *Realization problem for fractional continuous-time systems*, Archives of Control Sciences, vol. 18, no. 1, 2008, 43-58.

[Kaczorek, 2008c] T. Kaczorek. *Realization problem for positive 2D hybrid systems*, COMPEL, vol. 27, no. 3, 2008, 613-623.

[Kaczorek, 2009a] T. Kaczorek. *Fractional positive linear systems*. Kybernetes: The International Journal of Systems & Cybernetics, 2009, vol. 38, no. 7/8, 1059–1078.

[Kaczorek, 2009b] T. Kaczorek. *Polynomial and Rational Matrices*, Springer-Verlag, London, 2009.

[Kaczorek, 2011a] T. Kaczorek. *Computation of positive stable realizations for linear continuous-time systems*, Bull. Pol. Acad. Sci. Techn., vol 59, no. 3, 2011, 273-281 and Proc. 20th European Conf. Circuit Theory and Design, August 29 to 31, 2011, Linköping, Sweden.

[Kaczorek, 2011b] T. Kaczorek. *Positive stable realizations of fractional continuous-time linear systems*, Int. J. Appl. Math. Comp. Sci., Vol. 21, No. 4, 2011, 697-702.

[Kaczorek, 2011c] T. Kaczorek *Positive stable realizations with system Metzler matrices*, Archives of Control Sciences, vol. 21, no. 2, 2011, 167-188 and Proc. Conf. MMAR'2011, CD-ROM.

[Kaczorek, 2011c] T. Kaczorek. *Selected Problems in Fractional Systems Theory*, Springer-Verlag 2011.

[Kaczorek, 2012] T. Kaczorek. Existence and determination of the set of Metzler matrices for given stable polynomials, Int. J. Appl. Comput. Sci., 2012 (in Press).

[Shaker and Dixon, 1977] U. Shaker, M. Dixon. *Generalized minimal realization of transfer-function matrices*, Int. J. Contr., vol. 25, no. 5, 1977, 785-803.

Author Information

Tadeusz Kaczorek *– Bialystok University of Technology; Faculty of Electrical Engineering;*

Wiejska 45D, 15-351 Bialystok, Poland; e-mail: kaczorek@isep.pw.edu.pl

Major Fields of Scientific Research: The theory of systems and the automatic control systems theory, specially singular multidimensional systems, positive multidimensional systems and singular positive 1D and 2D systems

NATURAL LANGUAGE PROCESSING AND WEB MINING

APPLICATION OF SOCIAL ANALYTICS FOR BUSINESS INFORMATION SYSTEMS

Alexander Troussov, D.J. McCloskey

***Abstract**: Social networking tools, blogs and microblogs, user-generated content sites, discussion groups, problem reporting, and other social services have transformed the way people communicate and consume information. Yet managing this information is still a very onerous activity for both the consumer and the provider, the information itself remains passive. Traditional methods of keyword extraction from text based on predefined codified knowledge are not well suited for use in such empirical environments, and as such do little to support making this information more an active part of the processes to which it may otherwise belong. In this paper we analyse various use cases of real-time context-sensitive keyword detection methods using IBM LanguageWare applications as example. We present a general high-performance method for exploiting ontologies to automatically generate semantic metadata for text assets, and demonstrate examples of how this method can be implemented to bring commercial and social benefits. In particular, we overview metadata-driven semantic publishing on the BBC FIFA World Cup 2010 website and the applications for social semantic desktops.*

***Keywords**: data mining, natural language processing, recommender systems, social semantic web, graph-based methods.*

***ACM Classification Keywords**: H.3.4 [Information Storage and Retrieval]: Systems and Software – information networks; H.3.5 [Information Storage and Retrieval]: Online Information Services – data sharing.*

Introduction

The massive scale production of human oriented information presents individuals and organizations with serious problems imposed by the limitations of the HTML and associated text based knowledge containers ubiquitous on the web. In this paper we

examine the general problem through study of a concrete example for each case, challenges facing the individual information consumer, the activity centric environment Nepomuk-Simple, and the information centric organization, BBC's World Cup 2010 website.

The core focus of a leading news and media organization is on providing the best and most up to date information in the most consumable way for the expected audience. The amount of data now flowing through our electronic world means that even in a controlled environment where a fixed number of journalists report on a topic there is still a need to relate and integrate the information created by those journalists with the torrential flows from all other channels. The definition of the "best and most up to date information" has shifted radically since the advent of the internet and accelerated through the web 2.0 social media explosion. Similarly, the definition of "consumable" and the nature of the audience have made step changes with advances in recommender systems, web presentation and information visualization and the expectations of automated feed consumption and programmatic access for mashup and situational or application based re-integration of the information content. In this paper we show that for many of the most pressing issues the semantic web standards coupled with state of the art text analysis techniques can make it possible for the information we place on the web to become programmatically available.

Converting the textual output of the social web into programmatically accessible knowledge is a well recognized challenge today. The components of a successful strategy in addressing this challenge are, at a high level:

1. Text analysis – for extracting the spans of text which represent agents and situations or concepts and relationships.
2. A domain knowledge description system and language – to formally describe the semantics of the real world topics discussed in the human language text.
3. A knowledge base used to store both the domain description and the instance of facts discovered in the text and support further global analysis and query.
4. Semantic analysis – to apply logic and graph based methods to the knowledge graph stored in the knowledge base.

The BBC FIFA World Cup 2010 website in an exemplar in terms of standard web based presentation of multifaceted information. It presents general reference information about the event itself, such as details of teams, fixtures, venues, individual player profiles and also dynamic information such as up to date news and special features, commentator blogs, match reports, video links, and interesting statistics as the tournament progresses. The site is laid out in a simple and classic way: layered menus at the top, blogs down

the left, main pane featuring the focus document or story with related information links to the right and bottom. The challenge was to automate the aggregation of related content on this multidimensional canvass. Manual selection, curation and placement of such information had been the norm, this was hugely labour intensive and would not scale to the new levels of dynamic information creation at such an event.

The rest of the paper is organized as follows. In section 2 we provide an overview of the requirements for natural language processing (NLP) covering the use cases considered in this paper. We show how these requirements were taken into account in IBM LanguageWare tools.

In section 3 we outline the architecture of the hybrid recommender system in the activity centric environment Nepomuk-Simple. In section 4 we overview metadata-driven semantic publishing on the BBC FIFA World Cup 2010 website. Finally, section 5 describes the conclusions and future work.

Dynamic Semantic Tagging

In this section we describe the requirements for NLP covering the use cases considered in this paper, these requirements effectively determine our use of the term dynamic semantic tagging. We show how these requirements were taken into account in IBM LanguageWare tools.

2.1 Requirements

First of all, the goals which drive the application of NLP to a real world problem are often determined by the business model, increasingly these business models are represented by ontology. Put simply, the most important task for the NLP application is to determine how the terms mentioned in a text are related to business model, including term disambiguation and ranking the relevancy of the text to the concept from the semantic network of ontological concepts. The social context, modeled as a network, is a natural extension of the semantic networks which are formed from concepts represented in ontologies. It is possible to use such networks for knowledge based text processing. In fact, layering these different networks upon each other brings mutual benefits, since more contextual linkage becomes available between concepts which might otherwise have been isolated. The gains of additional context come at a price of a challenge to runtime performance of the software which will analyze the network. Furthermore, many use cases demand near real time rapid response from the analytics algorithms and scalability is a major concern. There are also non-functional requirements from the software engineering perspective such easy software maintenance and reuse.

2.1 Outline of the Procedure

The basic procedure of mining texts using graph-based methods has two major steps, as depicted on the Fig. 1: lexical analysis combined with mapping from text to ontology, reasoning how concepts mentioned in a text sit together.

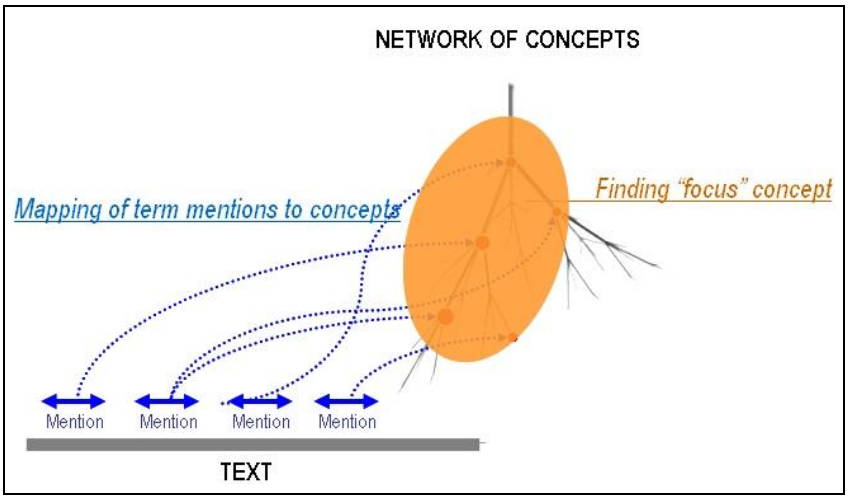


Fig. 1. Procedure of mining semantic models of texts consists of mapping from text to the ontology and reasoning how concepts mentioned in a text sit together with the objective of finding focus concepts.

Correspondingly, in the rest of this section we describe the approach taken by IBM LanguageWare to organise lexico-semantic resources for the purpose of mapping from text to ontology and for the selection of "reasoning" methods suitable for large networks.

2.3 Layered Dictionary Layout and Technical Organisation of Lexico-Semantic Resources into Lexically Enriched Ontology

According to [Ou et al., 2006] "It is an established fact that knowledge plays a vital role in the comprehension and production of discourse. The process of interpreting words, sentences, and the whole discourse involves an enormous amount of knowledge which forms our background and contextual awareness. However, how various types of knowledge can be organized in a computer system and applied to the comprehension process is still a major challenge for semantic interpretation systems used in natural language processing".

In LanguageWare a clear decision was made to separate lexico-semantic resource into two layers – lexical and semantic, and to provide a flexible framework for identifying term mentions of ontological concepts in raw text. Our pragmatic approach for representation of lexico-semantic resources is in vein with the most cited (in Computer Science and in

the field of Artificial Intelligence) Gruber's definition of ontology: "An ontology is an explicit specification of a conceptualization." [Gruber, 1993] with Guarino's clarifications [Guarino, 1998]: "… engineering artefact, constituted by a specific vocabulary used to describe a certain reality…". However, our goal is to design the means by which various types of knowledge can be organized in a computer system and applied to text processing for semantic annotation and IR applications.

The Fig. 2 shows IBM LanguageWare layered organisation of lexico-semantic knowledge.

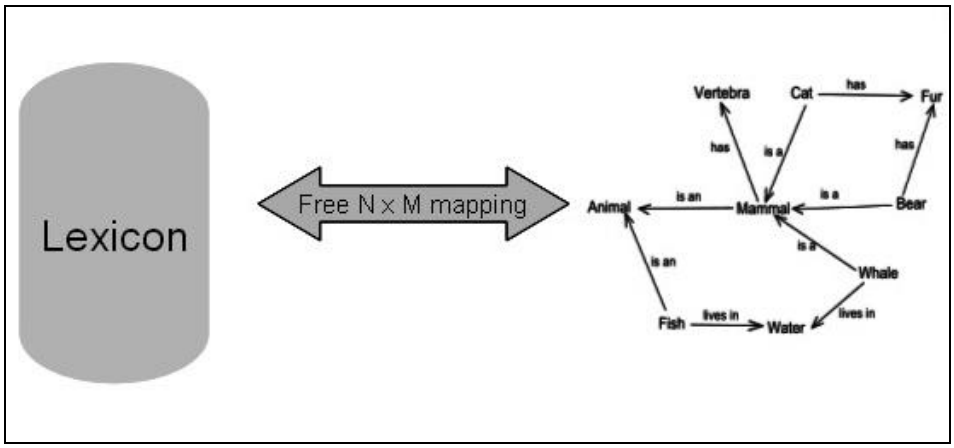


Fig. 2. Layered organisation of lexico-semantic knowledge for automatic text processing: lexical layer, semantic network layer, free N×M mapping between these two layers, lexical ambiguity phenomena (like synonymy and polysemy) are expressed by mapping

The Fig. 3 outlines processing resources which utilise these layers.

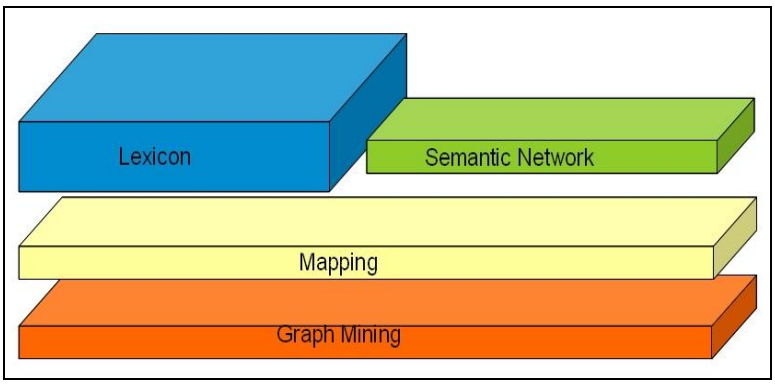


Fig. 3. Layered organisation and processing resources: Lexicon is used by lexical analyser to find mentions of concepts represented by nodes in the semantic network; mapping from text to concepts creates semantic model of a text (as a function on nodes of the network which shows how concepts are -related to text); graph-mining provides analytics on term mentions

2.4 Natural Language Understanding and Graph-based methods

From computational point of view natural language understanding could be considered as inferencing. For instance, based on a taxonomy of geographical locations one might infer that a text which mentions Malahide might be relevant to Canada since Malahide is a township in Elgin County, Ontario, Canada. However, terms are ambiguous (for instance, there is a location Malahide, Co. Dublin, Ireland), the knowledge encoded in ontologies is never "the truth, the whole truth, and nothing but the truth", and high precision mapping from lexical entries to concepts (that is, for instance, to detect that the hotel Paradis Gisenyi Malahide in Rwanda is not related to Canada or Ireland) is first of all prohibitively slow, and secondly, requires creation of hand-coded rules specific to the ontology in question.

At the same time, if we analyze how the concepts mentioned in a text sit together, one can reason that a text which mentions Malahide and Europe – is a little bit more likely to be about Ireland than about Canada, text which mentions Malahide and Clontarf – is more likely to be about Ireland than about Canada, and cohesive coherent text which mentions: Malahide, Mulhuddart, Lansdowne, Clontarf, Donabate (all these locations are in the capital of Rep. of Ireland) - is almost for sure about Dublin.

This type of "fuzzy" inferencing could be efficiently implemented using methods of "soft mathematics", in our case – graph-based methods.

Formally, solution of many network data mining tasks boils down to the following problem: given an initial function $F_0(v)$ on the network nodes, construct the function $F_{lim}(v)$ which provides the answer. In different domains the function F_0 could be referred to as the initial conditions, the initial activation, semantic model of a text, etc. In ontology based text processing, the initial function F_0 is the semantic model of a text w.r.t. to the knowledge: for instance, $F_0(v)=0$ if the concept v is not mentioned in the text, $F_0(v)=n$ if the concept v is mentioned *n* times. The function $F_{lim}(v)$ should show the foci of the text; for instance, $Argmax(F_{lim})$ is the most important focus of the text, while $F_{lim}(Argmax(F_{lim}))$ is the numerical value of the "relevancy". In information retrieval, the link analysis (such as Google's PageRank ([Brin and Page, 1998], [Langville and Meyer, 2006]) ranks web pages based on the global topology of the network by computing $F_{lim}(v)$ using the iterative procedure where the initial condition is that all web pages are equally "important" ($F_0(v) \equiv 1$).

Computationally efficient and scalable algorithms usually compute the function F_{lim} using iterations: on each iteration the value of $F_{n+1}(v)$ is computed depending on the values of the function F_n on the nodes connected to the node v. This is a very broad range of

algorithms including PageRank, spreading activation, computation of eigenvector centrality using the adjacency matrix.

Most of the mathematical algorithms behind such iterative computations are the "network flow" algorithms: they are based on the idea that something is flowing between the nodes across the links, and the structural prominence of nodes could be explained and computed in terms of incoming, outgoing and passing through traffic. Similar iterative computational schemes have been used for long time in finite element analysis to solve physical problems including propagation of heat, of mechanical tensions, oscillations, etc. Although finite element analysis automata usually perform on rectangular (cubic, etc.) grids, the extension to arbitrary networks is feasible [Troussov et al, 2011a].

In the IBM LanguageWare products described in this paper, we exploited the principal approach based on the computational scheme described above. The rationale for our choice of the graph-based method, which could be described as network flow method, could be understood by comparison with the mature area of applications of graph-based methods such as social network analysis. The task of determining "important" concepts based on the analysis of how the concepts mentioned in a text sit together, is a task of finding structurally important nodes in the network of concepts. In social network analysis, many traditional measurements of structural importance could be viewed in accordance with dynamic model-based view of centrality in [Borgatti, 2005] "that focuses on the outcomes for nodes in a network where something is flowing from node to node across the edges" [Borgatti and Everett, 2006].

Specific version of the network flow method used was spreading activation, which allows us to provide further optimisation of the performance by using bread-first search ([Troussov et al., 2009]). In addition to this principal graph-mining technique, we also exploited additional empirics, such as one sense per discourse [Judge et al., 2008].

Hybrid Recommender System in the Activity Centric Environment Nepomuk-Simple

In this section we outline the architecture of the hybrid recommender system in the activity centric environment Nepomuk-Simple (EU 6th Framework Project NEPOMUK) following [Troussov et al., 2008].

"Real" desktops usually have piles of things on them where the users (consciously or unconsciously) group together items which are related to each other or to a task. The "pile" based graphical user interface, used in the Nepomuk-Simple, imitates this type of data and metadata organisation which helps to avoid premature categorisation and reduces the retention of useless documents.

Metadata describing user data is stored in the Nepomuk Personal Information Management Ontology (PIMO). Proper recommendations, such as recommendations for additional items to add to the pile, apparently should be based on the textual content of the items in the pile.

Although methods of natural language processing for information retrieval could be useful, the most important type of textual processing are those which allow us to relate concepts in PIMO to the processed texts. Since any given PIMO will change over time, this type of natural language processing cannot be performed as preprocessing of all textual context related to the user. Hybrid recommendation needs on-the-fly textual processing with the ability to aggregate the current instantiation of PIMO with the results of textual processing.

Modeling this ontology as a multidimensional network allows the augmentation of the ontology with new information, such as the "semantic" content of the textual information in user documents. Recommendations in Nepomuk-Simple are computed on the fly by graph-based methods performing in the unified multidimensional network of concepts from PIMO augmented with concepts extracted from the documents pertaining to the activity in question.

In [Troussov et al., 2008] Nepomuk-Simple recommendations were classified into two major types. The first type of recommendations is the recommendation of additional items to the pile, when the user is working on an activity. The second type of recommendation arises, for instance, when the user is browsing the Web: Nepomuk-Simple can recommend that the current resource might be relevant to one or more activities performed by the user.

Application for Dynamic Semantic Publishing

The Internet is changing how news is being consumed, and news organizations need to respond with dynamic, responsive, timely, relevant, and quality online content. Content providers want to be able to dynamically compose new web pages in response to external events, to ensure semantic interoperability of their content, to improve navigation, to facilitate content re-use and re-purposing, to reduce overall publishing costs, and to introduce new publishing models at zero incremental effort to the journalists.

"BBC News, BBC Sport and a large number of other web sites across the BBC are authored and published using an in-house bespoke content management/production system ("CPS") with an associated static publishing delivery chain. …The first significant move away from the CPS static publishing model by the BBC's Future Media department was through the creation of the BBC Sport World Cup 2010 website" [Rayfield, 2012].

The BBC's FIFA World Cup 2010 website ([BBC World Cup 2010]) has more than 700 hundreds of topically composed pages for individual football entities such as football teams, groups and players aggregated via semantic web technologies. BBC on-line team coined a new technical term to describe this technology to automate the aggregation and publication of interrelated content objects - "Dynamic Semantic Publishing" [Nowack, 2010].

The usual bottleneck for proliferation of semantic web is the need for manual and tedious work for annotation. IBM LanguageWare, which is now part of IBM Content Analytics, is the text analysis technology that is being used in this project by BBC on-line team to overcome this bottleneck ([MacManus, 2012]).

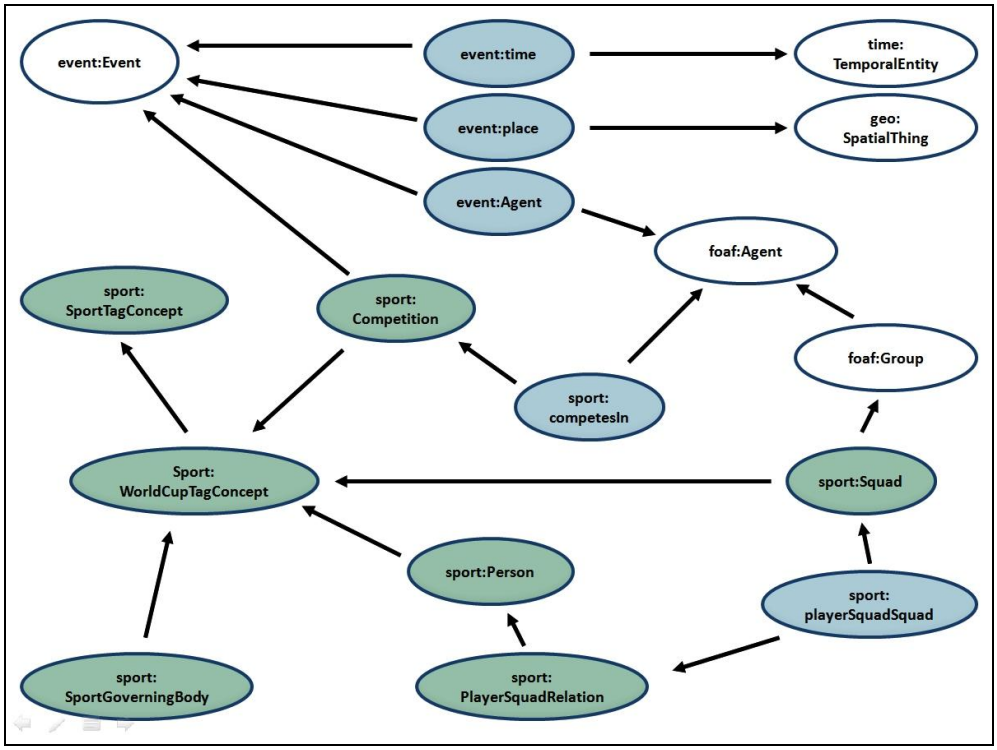


Fig. 4. Domain ontology used in the BBC FIFA World Cup 2010 Website [BBC World Cup 2010], simplified for brevity ([Rayfield, 2012])

The BBCs Fluid Operations' Information Workbench supports the editorial process for the BBC's Dynamic Semantic Publishing strategy, from authoring and curation to publishing of ontology and instance data following an editorial workflow [Zaino, 2012]. According to Peter Haase, Lead Architect R&D at Fluid Operations, this Information Workbench, as deployed by the BBC, "integrates and interlinks dynamic and semantically enriched

data in a central place. Approved content is then available for automatic publication on the website. The platform seamlessly integrates into already existing editorial processes and automates the creation and delivery of semantically enriched content." [Zaino, 2012]

The basis of this system was an ontology that described how World Cup facts related to each other. For example, "Frank Lampard" was part of the "England Squad" and the "England Squad" competed in "Group C" of the "FIFA World Cup 2010". The ontology also included "journalist-authored assets" such as stories, blogs, profiles, images, video and statistics.

IBM LanguageWare provides natural language processing of all related documents w.r.t. this knowledge to identify key concepts and provide named entities disambiguation. IBM Content Analytics analyzes news, as it is created by journalists, and identifies key concepts (explicit or inferred) to a high level of accuracy. "When a journalist writes a story, an athlete is surfaced in suggestions to tag – when someone no one ever heard of before wins a gold medal, he is immediately identified. " (Jem Rayfield, Lead Technical Architect for the News and Knowledge Core Engineering department of BBC, [Zaino, 2012]).

Conclusions and Future Work

We described the architecture and algorithms of ontology based semantic processing used by IBM LanguageWare in several projects and outlined two applications of this technology: one for individual information consumer (Nepomuk-Simple semantic desktop) and another for the information centric organisation (the BBC's World Cup 2010 website). This reusable approach could be also useful in other applications. For instance, in [Kinsella et al., 2008] it was used for social network analysis application such as navigation in the ego-centric networks. Real life applications show that LanguageWare implementation is efficient and scalable; the paper [Judge et al., 2007] described experimental results for graph-based part of the system: network flow operations needed to process a text using semantic networks with several hundred thousand concepts take 200msc on an ordinary PC.

Future work might require additional elaboration on the dictionary layout, on the dynamic update of lexical layer, and on various issues that arise when applying two-layer approach to languages more morphologically rich and linguistic complex than English (see [Troussov et al., 2009a]). Since the major graph-mining operation in IBM LanguageWare is based on a network flow methods, which is very broad class of algorithms, finding the most suitable algorithms for the specific tasks and the topology of the knowledge network is still a challenge.

Bibliography

[Ou et al, 2005] Ou, Weiqiang, Elsayed, Adel, and Hartley, Roger. Towards ontology-based semantic processing for multimodal active presentation. Games Computing and Creative Technologies: Conference Papers, 2005.

[Gruber, 1993] Gruber, T.R. A translation approach to portable ontologies. Knowledge Acquisition, 5(2):199-220, 1993.

[Guarino, 1998] Guarino N. Formal Ontology in Information Systems. In Proceedings of FOIS'98, Trento, Italy, 6-8 June 1998. Amsterdam, IOS Press. 3-15.

[Brin and Page, 1998] Brin, S. and Page, L. The Anatomy of a Large-Scale Hypertextual Web Search Engine. In: Seventh International World-Wide Web Conference (WWW 1998), April 14-18, 1998, Brisbane, Australia.

[Langville and Meyer, 2006] Langville, A.N. and Meyer, C. Google's PageRank and Beyond: The Science of Search Engine Rankings. Princeton University Press, 2006

[Troussov et al., 2011a] Troussov, A., Darena, F., Zizka, J., Parra, D., and Brusilovsky, P. Vectorised Spreading Activation Algorithm for Centrality Measurement. Acta univ. agric. et silvic. Mendel. Brun. Brno, 2011

[Borgatti, 2005] Borgatti, S. P. Centrality and network flow. Social Networks, 27, 2005, 1: 55–71.

[Borgatti and Everett, 2006] Borgatti, S. and Everett, M. A graph-theoretic perspective on centrality. Social Networks, 28(4):466–484, 2006.

[Troussov et al., 2009] Troussov, A., Levner. E., Bogdan, C., Judge, J., and Botvich, D. Spreading Activation Methods. In Shawkat A., Xiang, Y. (eds). Dynamic and Advanced Data Mining for Progressing Technological Development, IGI Global, USA, 2009.

[Judge et al, 2008] Judge, J., Nakayama, A., Sogrin, M., Troussov, A. Method and System for Finding a Focus of a Document. United States Patent 7870141, Filing Date: 02/26/2008.

[Troussov et al., 2008] Troussov, A., Judge, J., Sogrin, M., Bogdan, C., Lannero, P., Edlund, H., Sundblad, Y. Navigation Networked Data using Polycentric Fuzzy Queries and the Pile UI Metaphor. Proceedings of the International SoNet Workshop (2008), pp. 5-12.

[Rayfield, 2012] Rayfield, J. Sports Refresh: Dynamic Semantic Publishing. Retrieved June 30, 2012, from http://www.bbc.co.uk/blogs/bbcinternet/2012/04/sports_dynamic_semantic.html

[BBC World Cup 2010] BBC World Cup 2010 Website. Retrieved June 30, 2012, from http://news.bbc.co.uk/sport2/hi/football/world_cup_2010/default.stm

[Nowack, 2010] Nowack, B. Dynamic Semantic Publishing for any Blog (Part 1). 2010. Retrieved June 30, 2012, from http://bnode.org/blog/2010/07/30/dynamic-semantic-publishing-for-any-blog-part-1

[MacManus, 2012] MacManus, R. BBC World Cup Website Showcases Semantic Technologies. July 2010. Retrieved June 30, 2012, from http://www.readwriteweb.com/archives/bbc_world_cup_website_semantic_technology.php

[Zaino, 2012] Zaino, J. Sports are the Semantic Focus in Britain at the BBC and in Brazil at Globo. 2012. Retrieved June 30, 2012, from http://semanticweb.com/sports-are-the-semantic-focus-in-britain-at-the-bbc-and-in-brazil-at-globo_b29040

[Kinsella et al., 2008] Kinsella, S., Harth, A., Troussov, A., Sogrin, M., Judge, J., Hayes, C., and Breslin, J.G. Navigating and Annotating Semantically-Enabled Networks of People and Associated Objects. Why Context Matters: Applications of Social Network Analysis (T. Friemel, ed.), VS Verlag, 2008, ISBN 3531163280, 79-96.

[Judge et al., 2007] Judge, J., Sogrin, M., and Troussov, A. Galaxy: IBM Ontological Network Miner. Proceedings of the 1st Conference on Social Semantic Web (CSSW), September 26-28, 2007, Leipzig, Germany.

[Troussov et al., 2009a] Troussov, A., Judge, J., Sogrin, M., Akrout, A., Davis, B., and Handschuh, S. A Linguistic Light Approach to Multilingualism in Lexical Layers for Ontologies. SLT, vol 12, Polish Phonetics Association, ed. G. Demanko, K. Jassem, S. Szpakowicz

Authors' Information

Alexander Troussov *– Ph.D., IBM Dublin Center for Advanced Studies Chief Scientist. Dublin Software Lab, Building 6, IBM Technology Campus, Damastown Ind. Est., Mulhuddart, Dublin 15, Ireland; e-mail: arousso@ie.ibm.com*

Major Fields of Scientific Research: natural language processing, software technologies, network analysis

D.J. McCloskey *– NLP Architect, IBM Watson. Building 6, IBM Technology Campus, Damastown Ind. Est., Mulhuddart, Dublin 15, Ireland; e-mail: dj_mccloskey@ie.ibm.com*

Major Fields of Scientific Research: computational linguistics, natural language processing, semantic web applications

MACHINE TRANSLATION IN THE COURSE "COMPUTER TECHNOLOGIES IN LINGUISTICS" AT THE PHILOLOGICAL DEPARTMENT OF THE SAINT-PETERSBURG UNIVERSITY

Andrei Masevich, Victor Zakharov

Abstract: *Machine translation now left laboratories and became one of the practices of information service. A large number of free or partly free systems of machine translation became available in the net. Consequently, the task of their comparison and evaluation criteria arises. The paper describes the procedure of the estimation of the machine translation of text executed by different machine translation systems. The systems did translation into the Russian language of text (fragment "Communist manifesto" by Marx and Engels) from German, English and French. Students of the philological department of Saint-Petersburg State University systematized errors of translation, tried to determine the sources of errors, and considered the possibility (or impossibility) of their elimination.*

Keywords*: machine translation, evaluation, teaching*

ACM Classification Keywords*: I.2.7 Natural Language Processing*

Introduction

Machine translation (MT) now became one of the widely used services of the global network. The task of comparison and evaluation of numerous systems that appeared recently arises. The growing popularity of systems of machine translation systems, the constant improvement of them are factors which cause the need of including certain aspects of these technologies into the different training courses and first of all for the students of linguistics. In this paper we describe the approach to the teaching machine translation in at? the Philological Department of St. Petersburg University (the Chair of Mathematical Linguistics) within training course "Computer technologies in linguistics".

To evaluate the machine translation one can use different criteria. From the? point of view of user, the output text should first semantically correspond to the source text and it should be coherent and understandable to target language audience [Helmreich, Farwell 1996: 49. The development of MT evaluation principles is both an important research task and the need of user facing the problem of choice of the system.

The Consortium Translate4eu developed a protocol of the translation evaluation from the user's point of view. The protocol is based on the set of tests (working package 4). The users are asked to evaluate the adequacy of the translation and some other features of systems. The users compare translations of sentences selected at random. Users have a chance to compare their own evaluation and complex summary of evaluations of a system. Result of the summary will be available to any user.

The quality of MT systems is in many respects determined by solution of linguistic problems. That is why one of the approaches to the evaluation of a systems - this is the examination of their linguistic special features.

We describe the general methodology of the evaluation of the productions of various systems of machine translation, which includes comparison of translated texts and statistical analysis of errors of translation. The students of the department act as experts who execute the evaluation. The paper also contains description of the practical training tasks.

The essence of the approach

We assume that when starting the course students have already certain knowledge of the linguistic disciplines and the theory and current practice of machine translation.

The practical task consists of identifying, classifying and statistical analysis of errors of several systems of machine translation and development of suggestions for improvement of the system's performance.

For the training tasks we use the following Internet resources: the translating systems Google, Prompt, AltaVista and the service **http://itranslate4.eu/**, which provide simultaneous translations by seven systems Bing, Google, Lingenio, Prompt, SkyCode, Systran, Trident, also the National Corpus of Russian Language, and Electronic dictionary of German (Digitales Wörterbuch der deutschen Sprache (DWDS)

Using the systems, the students translate separate sentences and small fragments of text from English, German, and French into Russian. For the comparison one uses the human translations, executed by professional interpreters and sometimes students make translations themselves. Students further classify the errors of machine translators try to understand their reasons and consider possibilities of the corrections.

Classification of errors

For the classification of errors a rough scheme was developed. We divide errors of translation into three large groups. One can further divide each group into subgroups.

1. Lexical *and semantic errors*

Wrong translation of words and word-groups, for example:

- Bill has been out of work for several months (Билл был без работы в течение нескольких месяцев) Законопроект был без работы в течение нескольких месяцев (Homonymy of words Bill – a draft of law and Bill – personal name)
- She was writing letters all afternoon (После полудня она все время писала письма) Она писала помечает буквами все время после обеда (Homonymy letters written messages and letters characters)
- It's so nice sitting here with you (Так хорошо сидеть здесь с вами) Это такое хорошее заседание здесь с вами. (The gerund "sitting" should be translated with iinfinitive form)

2. Grammar Errors

I am going to Moscow next month (Я собираюсь в Москву следующем месяце)
Я иду к Москва следующий месяц – coherence error

To know her is to like her Знать ее полюбить она – coherence error

When he lived in London he went to the theatre once a week (Когда он жил в Лондоне, он ходил в театр каждую неделю) Когда он жил в Лондоне, он пошел в театр один раз в неделю – Predicate, expressing repeatable action, is translated as predicate of once occurring action

The earth moves round the sun (Земля движется вокруг Солнца) Земля двигает вокруг солнца. The relation agent –subject is rendered wrongly.

The sun shines during the day (Солнце светит днем)Блески солнца во время дня. The predicate is translated as subject

3. The other errors (generating of pseudo words, lack of translation of words and collocations etc.)

It was a rule for men and women to sit apart Было правилом для людей и *женщин*, котор нужно сидеть врозь

Studying all night made him ill Изучать всю ночу сделал его больно

When he lived in London he went to the theatre once a week (Когда он жил в Лондоне, он ходил в театр каждую неделю) Когда он жил в лондоне он пошел к театру once a week

Each type of error has a corresponding code 1, 2 or 3. The absence of error is designated by the 0

Students assign one or more codes to each translated sentence. The codes should be

inserted in brackets to the boxes of the table as shown bellow (see tabl.1).

Translation of sentences

To illustrate this task we selected 9 sentences from the book [Hornby, 1975] (see Table 1).

Table 1. Results of translation of sentences executed by 4 systems

English sentence	Promt	Trident	Systran	WordLINGVO
Studying all night made him ill	Изучение всю ночь сделало его плохо (2)	Изучение всей ночи сделало его больным (2)	Изучать всю ночу сделал его больно (2,3)	Изучать всю ночу сделал его больно (2,3)
It was a rule for men and women to sit apart	Это было правило для мужчин и женщин сидеть обособленно(2)	Это было правило, чтобы мужчины и женщины сидели отдельно (0)	Было правилом для людей и женщин, котор нужно сидеть врозь (1, 2, 3)	Было правилом для людей и женщин, котор нужно сидеть врозь (1, 2, 3)
When he lived in London he went to the theatre once a week	Когда он жил в Лондоне, он пошел в театр один раз в неделю (2)	Когда он жил в Лондоне, он ходил в театр раз в неделю (0)	Когда он жил в Лондоне он шел к театру раз неделя (2)	Когда он жил в лондоне он пошел к театру once a week (2,3)
She was writing letters all afternoon	Весь день она писала письма (1)	Она писала помечает буквами все время после обеда (1,2)	Она писала письмам все после полудня (2)	Она была письмами сочинительства все после полудня (2,3)
To know her is to like her	Знать ее означает понравиться она (2)	Чтобы знать ее - понравиться ее (2)	Знать ее полюбить она 2)	Знать ее к как ей (1,2)
Я иду в Москву в следующем месяце	Я иду в Москву в следующем месяце (2)	Я собираюсь в Москву в следующем месяце (0)	Я иду к Москве следующий месяц (2)	Я иду к Moscow следующий месяц (2,3)
The sun shines during the day	В течение дня светит солнце (0)	Солнце сияет в течение дня (1)	Блески солнца во время дня (2)	Солнце светит во время дня (0)

Table 1-continued. Results of translation of sentences executed by 4 systems

English sentence	Promt	Trident	Systran	WordLINGVO
Bill has been out of work for several months	Билл был безработным в течение нескольких месяцев (0)	Билл был безработным несколькими месяцами (2)	Билл из работы на несколько месяцев (2)	Билл из работы на несколько месяцев (2)
It's so nice sitting here with you	Это - так хорошее заседание здесь с Вами (1,2)	Это - такое хорошее заседание здесь с вами (1,2)	Оно настолько славный сидеть здесь с вами (1,2)	Будет настолько славным усаживанием здесь с вами(1, 2)

Table 2. Evaluation of translation according to types and number of errors

Type of error	Code of error	Promt, number of errors	Trident, number of errors	Systran, number of errors	World - LINGO, number of errors
Lexical and semantic errors	1	2	3	2	3
Grammar errors	2	6	5	9	8
Other	3	0	0	1	5
General number of errors		8	8	12	16
Number of sentences translated without errors	0	2	3	2	1

For the examples, we selected 9 English sentences. But even a small amount of material allows to come to certain conclusions, like following:

Predominated type of errors are grammar errors.

Since the Trident system produce the more errorless translations than any other one can suggest that the translation of this system are of the best quality.

The amount of material cold be enlarged and for its processing one can apply statistical methods.

Further, the students summarize the results obtaining in several experiments.

In addition to evaluation of machine translators, the students are asked to identify the grammar and syntax constructions especially difficult for machine translation from English into Russian. To perform this it is necessary to analyze sentences with most serious errors of translation.

Thus, judging by our example, one can suggest that one of the most complex for the machine translation from English into Russian is gerund constructions. The students are asked to find corresponding examples to confirm or to refute the suggestion as follows (Table 3).

Table 3. Translations of sentences with gerund

Sentences	Promt	Trident	Systran	WordLINGVO
She enjoys going to concerts	Она любит идти в концерты	Она наслаждаются собрался в концерты	Она наслаждается идя концертами	Она наслаждается пойти к согласиям
She can't bear seeing animals treated cruelly	Она не может родить видящих животных, которых рассматривают безжалостно	Она не может носить виденье, что животные лечили жестоко	Она не может принести увидеть животных обработанных жестокосердно	Она не может принести увидеть животных обработанных жестокосердно
Don't start borrowing money	Не начинайте занимать деньги	Не запускайте одолжение денег	Не начните одолжить деньгам	Не начните одолжить деньг

The analysis of the examples seems to confirm the suggestion. It also can help the regularities of the some errors appearance.

The translation of the fragment of the text

As another training task we take a translation by translating systems of the fragment from the Manifesto of Communist Party by K. Marx and F. Engels from German, English and French. The choice is because the versions of the text are available in Internet in all three languages. There is also Russian translation (we use the Russian version published by the Marx-Engels-Lenin Institute in 1955).

In our paper we limit ourselves with analysis of some errors in translating from German into Russian

Source text (German)

Ein Gespenst geht um in Europa - das Gespenst des Kommunismus. Alle Mächte des alten Europa haben sich zu einer heiligen Hetzjagd gegen dies Gespenst verbündet, der Papst und der Zar, Metternich und Guizot, französische Radikale und deutsche Polizisten.

Wo ist die Oppositionspartei, die nicht von ihren regierenden Gegnern als kommunistisch verschrien worden wäre, wo die Oppositionspartei, die den fortgeschritteneren Oppositionsleuten sowohl wie ihren reaktionären Gegnern den brandmarkenden Vorwurf des Kommunismus nicht zurückgeschleudert hätte?

Translation into Russian by the Marx-Engels-Lenin institute, 1955 г.):

Призрак бродит по Европе - призрак коммунизма. Все силы старой Европы объединились для священной травли этого призрака: папа и царь, Меттерних и Гизо, французские радикалы и немецкие полицейские.

Где та оппозиционная партия, которую ее противники, стоящие у власти, не ославили бы коммунистической? Где та оппозиционная партия, которая в свою очередь не бросала бы клеймящего обвинения в коммунизме как более передовым представителям оппозиции, так и своим реакционным противникам?

Translation into English

A specter is haunting Europe--the specter of Communism. All the powers of old Europe have entered into a holy alliance to exorcise this specter; Pope and Czar, Metternich and Guizot, French radicals and German police spies.

Where is the party in opposition that has not been decried as Communistic by its opponents in power? Where the opposition that has not hurled back the branding reproach of Communism, against the more advanced opposition parties, as well as against its reactionary adversaries?

Table 4. Examples of translating the fragment

Systems	Translation into Russian
Promt German - Russian	**Привидение (1)** идет вокруг в Европе - привидение коммунизма. Все власти (1) старой Европы объединились к святой (1) травле против этого привидения, папа и царь, Metternich и Guizot (3), французские радикалы и немецкие полицейские. Где оппозиционная партия, которая не была бы поносившей (2) ее правящими противниками как коммунистическая, где оппозиционная партия, у которой не было бы клеймящего упрека коммунизма zurückgeschleudert (3) более прогрессивным оппозиционным людям как ее реакционным противникам
Trident German - Russian	**Привидение** идет, чтобы в Европе - привидение коммунизма. Все власти старой Европе союзные себя к святому Hetzjagd против этого привидения, Papst и Zar, Metternich и Guizot, французский радикальный и немецкий полицейский. Где оппозиционная партия есть, которая не стала бы обесславлена их владетельными противниками как коммунистический, где оппозиционная партия, которая не назад бросила бы fortgeschritteneren оппозиционным людям как и их реакционным противникам клеймящий упрек коммунизма
Promt German - English, Systran English- Russian	**Привидение** идет вокруг в Европу - привидение коммунизма. Вся власть старой Европы формировали союзничество к святейшей охоте против этого привидения, Папы и царя, Metternich и Guizot, французских радикалов и немецких полицейскиев. Где оппозиционная партия которая не была бы обозначена ее господствующими противницами как коммунист, где оппозиционная партия которая не имела бы упрек zurückgeschleudert коммунизма к людям противовключения fortgeschritteneren как к ее охотнорядческим противницам
Lingatec German - English, Trident English- Russian	**Призрак** циркулирует в Европе - призрак коммунизма. Союзнический, римский папа и Царь, Metternich и Guizot, имеют себя для священной охоты с гончими против этой ghost Европы все полномочия века, Французских радикалов и Немецких полисменов. Где оппозиционная партия, оппоненты, не управляющие над этим hers, получены были печально известными как коммунист, где оппозиционная партия, которая не сможет иметь обеих обратных сторон, бросила клеймящий упрек коммунизма для более передовых оппозиционных людей подобно ее реакционным оппонентам

Table 4 continued. Examples of translating the fragment

Systems	Translation into Russian
Lingemo German - English, Promt English- Russian	**Призрак** циркулирует в Европе, призраке коммунизма. Все полномочия старого человека Европа соединились с собой для священной стремительной охоты против этого призрака, Папы Римского и Царя, Metternich и Guizot, французских радикалов и немецких полицейских. Где оппозиционная партия, неуправление противников по этому ее было столь коммунистическим печально известно оппозиционная партия, которые оба не бросились бы назад, упрек объявления коммунизма для более продвинутых людей оппозиции как ее реакционные противники будет иметь, где
Lingemo German - English, Trident English - Russian	**Призрак** циркулирует в Европе, призрак коммунизма. Все полномочия старой Европы особы имеют вступил себя в союз для священного разбивания, охотничьего против этого призрака, римского папы и Царя, Metternich и Guizot, Французские радикалы и Немецкие полисмены. Где оппозиционная партия, оппоненты, неуправляющие над этим hers, были как коммунист общеизвестно оппозиционная партия, которая не сможет бросить оба назад клеймящий упрек коммунизма для более передовых оппозиционных людей подобно ее реакционным оппонентам имел бы, где
Google German - English, Trident English - Russian (2010)	А **спектр** является навязчивый Европе - призрак коммунизма. Все державы старой Европы взяли на себя обязательства по Святым Духом охоту против этого альянса, Папы Римского, и царь, Меттерних и Гизо, французские радикалы и немецкие полицейские. Где оппозиционной партии, а не его оппонентов, как правящая Коммунистическая verschrien был обнаружен, когда оппозиционная партия, оппозиция более продвинутых людей, как своих реакционных противников, таких, как марка упрек коммунизма zurückgeschleudert бы и нет
Google German - Russian (2012)	**Призрак** бродит по Европе - призрак коммунизма. Все силы старой Европы объединились для священной травли этого призрака: папа и царь, Меттерних и Гизо, французские радикалы и немецкие полицейские. Где та оппозиционная партия, партия, которая не была осуждена своих противников как коммунистической правящей, где оппозиционная партия, которая не бросала бы более продвинутый оппозиции, а также своим реакционным противникам брендинга обвинения в коммунизме?

Semantic problems. We shall show one of semantic errors. Let us consider semantics of the Russian word “Призрак” which translated in versions 1 and 2 as “привидение”

Students are asked to look at the very detailed description of the lexical unit “Gespenst” in the electronic dictionary of German language (Digitales Wörterbuch der deutschen

Sprache (DWDS) **http://www.dwds.de/?qu=Gespenst**) (Fig.1) and to make a search for the examples of using words «призрак» and «привидение» in National Corpus of Russian Language.

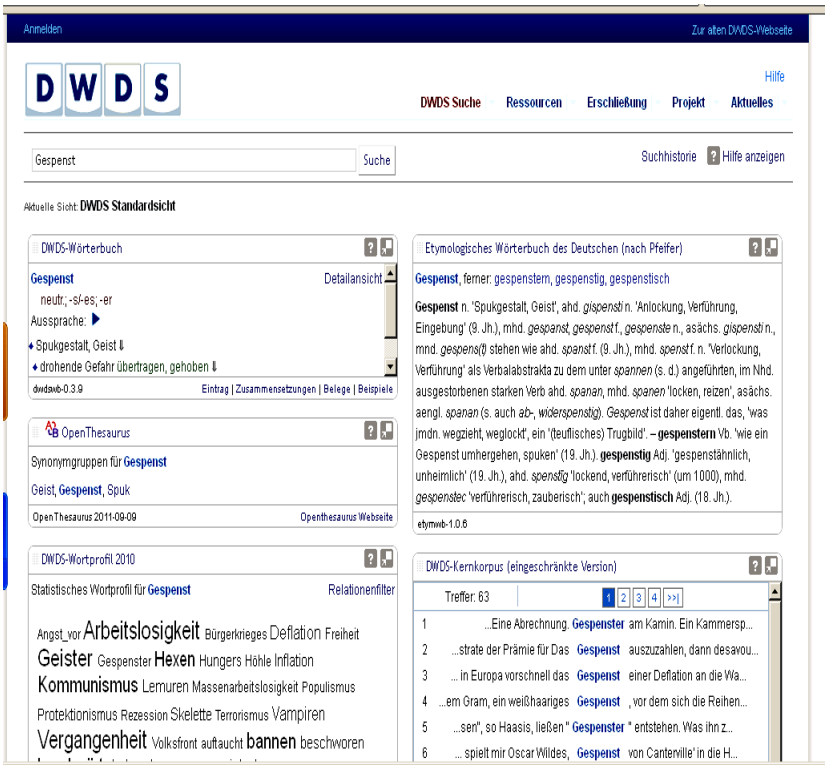


Fig. 1. Description of the word «Gespenst» in DWDS

The words «призрак» and «привидение» in Russian are synonyms. However, the word «призрак» is polysemic, it can mean both hope, threat, apprehension etc. and the host, mystical creature. The same is true for the German word “Gespenst”.

We selected several example of sentences with word «призрак» with different meanings from the National Corpus of Russian Language.

*На горизонте появился **призрак** войны с Турцией, вырисовывались контуры большой морской войны. [Е. В. Тарле. Экспедиция адмирала Сенявина в Средиземное море (1805-1807) (1954)]*

*Она дала мне хоть иллюзию, хоть **призрак** любви, и это истинно царский, неоплатный подарок… [А. И. Куприн. Телеграфист (1911)]*

Второе: суровая жизнь закалила Диму и Геню до состояния стали, а Егорушка рос

мягкий, ласковый, как бы безвольный, и уже виделся за горами ***призрак*** *его возможного будущего, будущего всех мужчин Диминой семьи, и история зарождения Егорушки сулила также и со стороны матери легкомыслие и случайные связи. [Людмила Петрушевская. Два бога (1998-1999)]*

То был ***призрак*** *короля Арнульфа Второго, изменнически, из засады убитого в 1527 году, на пятнадцатом году жизни. [Ф. К. Сологуб. Королева Ортруда (1909)]*

And also three phrases from the Corpus with the word “Привидение”

Охваченный состраданием, я хочу подойти к ней, но нимфа быстро поднимается: она только взглянула на меня, — и убегает, а на лице её ужас, точно она увидела ***привидение****". [И. Ф. Анненский. Вторая книга отражений (1909)]*

И ***привидение****, пройдя в отверстие трельяжа, беспрепятственно вступило на веранду. [М. А. Булгаков. Мастер и Маргарита, часть 1 (1929-1940)]*

Тут все увидели, что это— никакое не ***привидение****, а Иван Николаевич Бездомный — известнейший поэт. [М. А. Булгаков. Мастер и Маргарита, часть 1 (1929-1940)]*

Polysemy of the word “призрак” is evident that the examples above (ex.1-4). The word “привидение” is monosemic (ex.5-7) and its single meaning coincide with meaning of the word “призрак”, in which it is used in the ex.4

The word “Gespenst” in the source text joins metaphorically both meanings. On one side it is threat, apprehension. On the other side it is a subject of “heilige Hetzjagd”, i.e. of chase, persecution. In the literary translation of the text this metaphor is fully kept. But how should one set a task o this kind to the system of machine translation? Possibly, though we are not sure, when the system have to make a choice between two synonyms it should prefer the more general one.

Table 5. The example of Grammar error (from the version 1)

Wo ist die Oppositionspartei, die nicht von ihren regierenden Gegnern als kommunistisch verschrien worden wäre…	Где оппозиционная партия, которая не была бы поносившей ее правящими противниками как коммунистическая…

Should the participle «поносившей» (abusing, denouncing) be in the passive voice like this «Где оппозиционная партия, которая не была бы поносимой ее правящими противниками, как коммунистическая». However, it does not sound euphonic, so it is might be better to replace it with synonym «ославленной».

Errors of the third group are obvious. The personal names and the complex German words as zurückgeschleudert or fortgeschritteneren are frequently remaining not

translated.

Other particular qualities. In the version 7 the word “Gespenst” is translated as “спектр”. Neither in Russian nor in German (Spektrum) the word is used in the meaning of «призрак». The cause is that the system used an intermediate language. The text was first translated from German into English and then from English into Russian. The system translated English word “specter” as “спектр”, which has completely different meaning. Note, that the Google translator executed the version 7 in 2010.

The same system executed The version 8 was done in 2012. The first paragraph of this version copies literally Russian literary translation. The second paragraph is the usual machine translation with typical errors

Conclusion

In our paper, we could not give detailed analysis of all errors. We just had a task describe method of the evaluation of the system using the rough scheme of the classifying errors.

In conclusion, we would like to note that one can use our method could for other goals. If we refine the classification of errors than with the aid of students-linguists, we could possibly create a database of errors for improving systems of machine translation.

Bibliography

[Hornby, 1975] A.S.Hornby. Guide to patterns and usage in English London: Oxford University Press, 1975.

[Helmreich, Farwell, 1996] Helmreich St., Farwell D. Translation Differences and Pragmatic-Based MT //Expanding MT Horizons. Proc. of the Sec. Conf. Of the Assoc. for MT in the Americas. Montreal, Canada, 1996. Pp. 66-75.

Authors' Information

Andrei Masevich *– Assistant professor. Saint-Petersburg State University of Culture and Arts; e-mail: andmasev@mail.ru*

Major Fields of Scientific Research: Computational Linguistics, Library information systems

Victor Zakharov *– Associate Professor, Saint-Petersburg State University, Universitetskaya emb., 11, Saint-Petersburg 199034 Russia ; e-mail: vz1311@yandex.ru*

Major Fields of Scientific Research: Natural Language Processing

CLASSIFICATION OF PRIMARY MEDICAL RECORDS WITH RUBRYX-2: FIRST EXPERIENCE

Olga Kaurova, Mikhail Alexandrov, Ales Bourek

***Abstract**: RUBRYX is a document classifier developed in 2000s for processing large volumes of Web information. RUBRYX uses weighted sum of n-grams (n=1,2,3) extracted from a very limited number of samples (about 5-10) and takes into account their mutual position in a given text. This sophisticated algorithm proves to be very effective in classifying primary medical records presented in a free text form. In the paper we study possibilities of RUBRYX (version 2.2) on a limited document set in Spanish. These documents are medical histories related to stomach diseases. Such area should be considered as a narrow subset of medical records. The high quality of archived results (accuracy 80%-90%) allows us to recommend RUBRYX for similar applications.*

***Keywords**: natural language processing, medical diagnostics, document classification*

***ACM Classification Keywords**: I.2.7 Natural Language Processing*

Introduction

1.1 Problem setting

The subject under consideration is classification of primary medical records presented in a free text form as usually produced by medical professionals. Each document used here is related to a certain disease. So, in this case the medical record classification can be considered as a means of document based medical diagnosis decision support. The solution of this problem allows:

- To monitor medical doctors responsible for primary medical observation and to help reduce medical errors;
- To facilitate data exchange between different medical centers and to coordinate the storage and retrieval of individual records with the aid of computers;
- To help form Internet communities with similar health issues areas of interests.

The first significant publications in the addressed area appeared almost 20 years ago. The authors used Bayesian classifiers for processing encounter notes [Aronov, 1995a;

Aronov, 1995b]. An interesting and comprehensive work was published in 2006. It demonstrated that in spite of the use of advanced algorithms of classification, such as the SVM, the results prove to be not so good [Rost, 2006]. We assume that this can be explained by a weak application of lexical resources to the documents under consideration. In a recent publication [Zhang, 2010] the authors use structural patterns in encounter notes, which allowed to improve the results. A short review concerning classification of free text clinical narratives was published last year [Kaurova, 2011]. It contains a description of some medical corpora, methods and software tools. The results described in this review led us to find and test new algorithms in order to improve existing results without the need for additional extraordinary efforts.

In the paper we study possibilities of the document classifier RUBRYX to process such specific documents as primary medical records. In the experiments we use the last version of the mentioned program. The version 2.2 is free shareware and can be easy downloaded [Rubrix, http]. The document set includes 55 documents related to 6 stomach diseases. It allows for analyzing the results of experiments in detail.

The RUBRYX algorithm uses patterns in the form of one-word terms, bigrams and trigrams and takes into account their joint position in a document. Currently are not aware of any publications describing medical records classification by RUBRYX. The above mentioned circumstances define the objectives of our work.

1.2 State of the art

Classification procedures are traditionally included into the technologies of Machine Learning and Data Mining. Well-known resources [Mitchell,1997; Bishop, 2006] provide good theoretical basis for the area. Document classification is covered in the books [Baeza,1999; Manning, 1999]. Here special attention is given to document indexing – the transformation of free text documents into their numerical form. A recent example of a text book containing many algorithms of document classification is [Manning, 2009].

There are many software packages on the market related to Machine Learning and Data Mining, for example: Weka [Weka, http], Rapid Miner [RapidMiner, http], CLUTO [CLUTO, http], R [R, http]. Some of these have ad-hocs for working with textual documents [WekaText, http; RText, http]. These ad-hocs use very simple procedures of text indexing that can not, and do not, give satisfactory results.

The program RUBRYX was developed in 2000s [Polyakov, 2001; Polyakov, 2003]. This program proved to be very friendly for end-users because of its simplicity in training and tuning. RUBRYX demonstrated its advantages on the famous set of Reuter news. Namely it provided the *F*-measure of 86% with only 5 representatives from each of 10 categories used for training. Other algorithms could reach the levels of 75%-92% of *F*-measure using

dozens of documents for training. One should mention here an excellent work where these results are shown [Stein, 2003a].

In section 2 we describe lexical resources and classification algorithm of RUBRYX. In section 3 we present the corpus used in our study. In section 4 we present the results of experiments. Short discussion is provided in section 5. Section 6 contains conclusions.

RUBRYX description

2.1 Training (preprocessing)

We present RUBRYX description because we could not find it in literature. Hereinafter we use the following terminology. By 'mini-vocabulary' we mean a vocabulary related with a concrete category. These mini-vocabularies are created during the training stage. By 'terminological vocabulary' we mean a vocabulary created for a given domain by external experts. This vocabulary reflects a common terminology for all categories in a document corpus. Terminological vocabulary is not obligatory for RUBRYX functionality. RUBRYX can create its mini-vocabularies with the support of terminological vocabulary as well as without it.

Both mini-vocabulary of a given category and a common terminological vocabulary contain 3 lists:

- one-word terms
- two-word terms (bigrams)
- three-word terms (trigrams)

To create mini-vocabularies a user selects several of the most representative documents from each category. Let us have *M* documents related with a certain category. The procedure consists in the following:

- All stop terms are eliminated
- All common one-word terms form the first list in the file WordList
- All common bigrams form the second list in the file WordLst2
- All common trigrams form the third list in the file WordLst3

Speaking 'common' we mean terms which occur at least in *m* documents, here $m \le M$. In our experiments we set $m=M$. The terminological vocabulary is an additional filter for term selection. Namely, RUBRYX selects those terms from WordList, WordLst2, and WordLst3, which occur also in the terminological dictionary.

The procedure presented above is implemented for all *n* categories. Therefore if we have *n* categories then the result of preprocessing will be 3*n* lists of terms.

Stop terms have their own vocabulary. This vocabulary consists of 3 lists with one-word terms, bigrams and trigrams respectively. The titles of files are fixed as: BlackList, BlackList2, BlackList3. Unlike the mentioned mini-vocabularies and terminological vocabulary the black lists can use so-called 'regular expressions' [Expressions, http]. For example, '?' and '??' mean all words with one or two letters. The expression '*[0-9]*' means all words, which contain at least one number. Etc.

2.2 Classification (processing)

Algorithm

RUBRYX uses the well known lineal algorithm: it calculates contribution of each category to a given document as a linear combination of category indexes [Baeza, 1999; Manning, 2009]. In our case the indexes are terms from the mini-vocabularies. Then the category having the largest contribution is announced to be a winner. Here is the short description

Let j be the number of category; $\{L_{j1}, L_{j2}, L_{j3}\}$ be the numbers of terms from all three lists in a given document; $\{N_{j1}, N_{j2}, N_{j3}\}$ be the numbers of all one-word terms, bigrams and three-grams in a given document. The contribution of j-category is:

$$C_j = K_1 (L_{j1}/N_{j1}) + K_2 (L_{j2}/N_{j2}) + K_3 (L_{j3}/N_{j3})$$

where $K_1+K_2+K_3 =1$. Obviously, that $C_j = [0, 1]$ for all categories

RUBRYX developers set the following values for K_i : $K_1=(0.2)/3$, $K_2= (1.3)/3$, and $K_3= (1.5)/3$. It is easy to see that $\sum_i K_i = 1$. These values were determined empirically on the basis of numerous experiments of the authors with different document sets. We use the same values in our research

Modifications

1) Thresholds for category selection

Traditional algorithm uses the following rule for decision making in case of hard classification:

the category j is a winner if $C_j=max_i (C_i)$.

RUBRYX uses a more complex rule, which takes into account the results of training. Namely, let T_j be a threshold for j-category. That is:

- a document belongs to j-category if $C_j \geq T_j$
- a document does not belong to j-category if $C_j < T_j$

But what to do when we have more than one satisfied condition, let for example, $C_1 \geq T_1$, $C_2 \geq T_2$. In this case the rule of decision making considers the values $\lambda_1= C_1 - T_1$, $\lambda_1= C_2 - T_2$. The category having the maximum λ-value will be the winner: $\lambda_j =max_i (\lambda_i)$

If all thresholds are too high and we have not even one satisfied condition $C_j \geq T_j$ then a given document can not be classified.

The thresholds { $T_1, T_2, \ldots T_n$ } are calculated during the training stage as a result of optimization problem solution. Namely, the best values of T_j provide the minimum number of errors when we classify the training document set.

2) Taking into account the term positions in a document

The developers suppose that terms better support their category when they are located together. For this reason the developers increase the weights of close terms in a document on a certain value *p.* Speaking of 'weights' we mean coefficients K_1, K_2, and K_3 introduced above. Speaking of 'close terms' we mean the simultaneous term occurrences in a given window. The developers fixed the window size S=10. Parameter p is an algorithm parameter, we set p=0.3. It is the advice of the developers. This value can be easy changed by a user.

2.3 Tuning

Having obtained the initial results of classification a user can change parameters of the algorithm to improve these results. The principal parameters to be changed are the thresholds T_j. Let we deal with *j*-category and let '*j*-documents' mean the documents from this category selected by RUBRYX.

Case 1: There are a *small* number of *j*-documents and a *small* number of alien documents.

> We decrease the threshold T_j that allows to increase the number of *j*-documents.
>
> The number of alien documents is expected not to increase in the same proportion.

Case 2: There are a *small* number of *j*-documents and a *large* number of alien documents.

> The situation is undefined. One should 'play' with the thresholds including the threshold T_j

Case 3: There are a *large* number of *j*-documents and a *small* number of alien documents.

> It is just what we want and we do nothing

Case 4: There are a *large* number of *j*-documents and a *large* number of alien documents.

> We increase the threshold T_j that allows to decrease the number of alien documents.
>
> The number of *j*-documents is expected not to decrease in the same proportion.

We use these rules in our experiments to tune RUBRYX

Experimental material

3.1 Corpus under consideration and its lexical resources

The corpus for this study is a collection of 55 anonymous primary medical records from one Clinical Hospital. The records are related to gastrointestinal diseases. Each of these records contains a short description of chief complaint, past medical history (including major illnesses, any previous surgery/operations any current ongoing illness, e.g. diabetes)., family history, medications, allergies, objective status and finally two diagnoses, the principal one and the concomitant one (morbidity and co-morbidity). Texts belong to 6 classes – diseases. The corpus is described in Table 3.1. Appendix presents an example of a primary medical record.

Table 3.1 Categories presented at the corpus

Class	*Disease*	*Number of texts*	*Number of words in all texts*	*Number of different words in all texts*
1	gallbladder disease	12	2849	428
2	mechanical jaundice	8	2076	458
3	stomach cancer	11	2873	572
4	acute appendicitis	6	1339	245
5	gastrointestinal bleeding	7	1525	373
6	inguinal hernia	11	2396	243
Total		55	12828	1269

3.2 Terminological vocabulary

Terminological vocabulary is not the obligatory element for RUBRYX work. But in many cases it can improve the quality of mini-vocabularies and as a consequence of the quality of classification.

The terminological vocabulary was constructed by external expert. It was the surgery related with the corpus of medical records described above. To construct the vocabulary we used the following technology:

Step 1.

All specific one-word terms were extracted from the whole corpus and presented to the expert. The expert selected the most interesting terms from this list. Here we used the program LexisTerm [Lopez, 2011].

Step 2.

All collocations (the left and right ones) with the selected terms were extracted from the corpus and presented to the expert again. He corrected the list and formed the final list of one-word terms, bigrams and trigrams. Here we used an auxiliary program.

For reasons of clarity we include following a brief comment concerning the program LexisTerm we used on the Step 1. For this we give two definitions [Lopez, 2011]:

Definition 1. The general lexis is a frequency word list based on a certain corpus of texts.

The 'certain corpus' means here any standard document set reflecting the lexical richness of a given language. Generally such a corpus contains in a certain proportion the documents taken from newspapers, scientific publications related with various domains, novels and stories. In our case it was the British National corpus.

Definition 2. The level of specificity of a given word **w** in a given document corpus C is a number $K \geq 1$, which shows how much its frequency in the document corpus $f_C(\boldsymbol{w})$ exceeds its frequency in the general lexis $f_L(\boldsymbol{w})$:

$K = f_C(\boldsymbol{w}) / f_L(\boldsymbol{w})$

Obviously, the more K is, the less words appear in the resulting list. We tested LexisTerm with K= {5,10, 20, 50,100}. From the expert point of view the value K=10 proved to be the best one.

Experiments

4.1 Measures for results evaluation

To evaluate the quality of classification we use several well-known measures being popular in Information Retrieval [Baeza, 1999]. Here is a short review of these measures:

Let a classifier selects l documents from the existing m documents to be selected, and let there are k really corrected documents between these n. In this case we can calculate Precision (P) , Recall (R), and F-measure using the formulae: $P= k/l$, $R= k/m$, $F=2PR / (P+R)$

Obviously these formulae refer to a binary classification. When we deal with several categories one should use the combined F-measure proposed in [Stein, 2003b]. We can use here also the traditional Accuracy: $A= n/N$, where n is a number of all correct classified documents, N is a number of all documents. Good survey of measures used in classification problems is presented in [Pinto, 2008].

4.2 Sensitivity to thresholds

Hereinafter we will use the following terminology. By 'class' we mean the Gold Standard. 'Category' is a result of classification with RUBRYX.

Table 4.1 Classification, *K* are calculated automatically

Class	*Training set*	*Test set*	*K=*	*1*	*2*	*3*	*4*	*5*	*6*	*Correct docs*
1	5	7	27	**1**	5			1		1
2	5	3	24		**3**					3
3	5	6	23		1	**5**				5
4	5	1	37				**1**			1
5	5	2	27					**2**		2
6	5	6	35						**6**	6
Total	30	25		1	9	5	1	3	6	18

Table 4.1 shows that Category 1 practically does not contain the documents from Class 1. According the rules described in the section 2.3 we decrease the threshold K_1 by 25%. Now it is K_1 =21. The results are presented in Table 4.2. We have here *Accuracy* = (18/25)*100% = 75%

Table 4.2 Classification, K_1 = 21 (25% decreased)

Class	*Training set*	*Test set*	*K=*	*1*	*2*	*3*	*4*	*5*	*6*	*Correct docs*
1	5	7	21	**7**						7
2	5	3	24		**3**					3
3	5	6	23		1	**5**				5
4	5	1	37				**1**			1
5	5	2	27					**2**		2
6	5	6	35	6					**0**	0
Total	30	25		13	4	5	1	2	0	18

In this experiment we use mini-vocabularies constructed on the basis of 5 samples *without a terminological vocabulary*. The results are presented in Table 4.1. Table rows contain the distribution of documents from a given class between categories. Table columns contain the distribution of all documents assigned to a given category between classes.

The accuracy is calculated as a ratio of the number of correct cases to the total number of cases. It is easy to see that *Accuracy* = (18/25)*100% = 75%

Table 4.2 shows that Category 6 is empty and Category 1 contains all documents of Classes 1 and 6. We have to go back to the original value of K_1=27 and to increase the threshold K_2 by 25%. Now it is K_2 =30. The latter was done to filter the documents of Class 1, which were put to Category 2 with K_1=27, see Table 4.1. The results of experiment are given in Table 4.3. We have now *Accuracy* = (23/25)*100% = 92%.

Table 4.3 Classification, K_2 = 30 (25% increased)

Class	***Training set***	***Test set***	***K=***	***1***	***2***	***3***	***4***	***5***	***6***	***Correct docs***
1	5	7	27	**5**		1		1		5
2	5	3	30		**3**					3
3	5	6	23			**6**				6
4	5	1	37				**1**			1
5	5	2	27					**2**		2
6	5	6	35						**6**	6
Total	30	25		5	3	7	1	3	6	23

As the result of the last experiment was successful we decide to complete the last experiment with K_1=24 and K_2=30. The results are given in Table 4.4. For this case *Accuracy* = (23/25)*100% = 92%.

Table 4.4 Classification, K_1=24 (10% decreased), K_2=30 (25% increased)

Class	***Training set***	***Test set***	***K=***	***1***	***2***	***3***	***4***	***5***	***6***	***Correct docs***
1	5	7	24	**6**				1		6
2	5	3	30		**3**					3
3	5	6	23			**6**				6
4	5	1	37				**1**			1
5	5	2	27					**2**		2
6	5	6	35	1					**5**	5
Total	30	25		7	3	6	1	3	5	23

4.3 Sensitivity to terminological vocabulary

Here we test the sensitivity of results to application of terminological vocabulary. Basically, we repeat the first experiment of the previous series of experiments, but now we use the vocabulary on the stage of training. The results are presented in Table 4.5. Here we have *Accuracy* = (21/25)*100% = 84%

Table 4.5 Classification with terminological vocabulary, *K* are calculated automatically

Class	*Training set*	*Test set*	*K=*	*1*	*2*	*3*	*4*	*5*	*6*	*Correct docs*
1	5	7	22	**5**	2					5
2	5	3	16		**3**					3
3	5	6	10			**5**		1		5
4	5	1	23				**0**	1		0
5	5	2	5					**2**		2
6	5	6	33						**6**	6
Total	30	25		5	5	5	0	4	6	21

Table 4.5 shows that the accuracies for Categories 2 and 5 are low, namely 60% for Category 2 and 50% for Category 5. So, we increase the thresholds for these categories by 25% and round it up in a higher number. We have now K_2= 20 and K_5 = 7. The results are given in Table 4.6. It is seen that *Accuracy* = (24/25)*100% = 96%.

Table 4.6 Classification with terminological vocabulary, K_2=20 and K_5=7 (both increased by 25%)

Class	*Training set*	*Test set*	*K=*	*1*	*2*	*3*	*4*	*5*	*6*	*Correct docs*
1	5	7	22	**7**						7
2	5	3	20		**3**					3
3	5	6	10			**6**				6
4	5	1	23				**0**	1		0
5	5	2	7					**2**		2
6	5	6	33						**6**	6
Total	30	25		7	3	6	0	3	6	24

4.4 Sensitivity to size of training set

For this series of experiments we use the most numerous classes. According to the Table 3.1 it is classes 1, 3 and 6. With these classes we can train and test RUBRYX on the largest number of samples. In the first experiment we use 3 documents for training from each class and in the second experiment we use 6 documents from each class. Naturally, the test set is the same in both cases. Terminological vocabulary is not used. *K*-values are calculated automatically. The results of classification are presented in Table 4.7 and 4.8 respectively.

To evaluate the quality of these results we use two measures: accuracy and combined *F*-measure. For the first experiment we have *Accuracy* = 75%, *F-measure* = 73%. For the second experiment we have *Accuracy* = 88%, *F-measure*=87%

Table 4.7 Classification with training set of 3 samples

Class	***Training set***	***Test set***	***K=***	***1***	***3***	***6***	***Correct docs***
1	3	6	32	**2**	4		2
3	3	5	30		**5**		5
6	3	5	42			**5**	5
Total	9	16		2	9	5	12

Table 4.8 Classification with training set of 6 samples

Class	***Training set***	***Test set***	***K=***	***1***	***3***	***6***	***Correct***
1	6	6	25	**4**	2		4
3	6	5	22		**5**		5
6	6	5	34			**5**	5
Total	18	16		4	7	5	14

Discussion

The potential possibilities of classification - any classification - are defined by relations between categories. The closer their characteristics are, the lower level of results we obtain. In case of document classification the closeness between categories is mainly defined by the intersection of lexis related with each category. When this intersection is absent the quality of classification is the highest one: we can avoid any errors. But when lexical resources of categories are similar then we can expect many errors. These extreme cases present so-called wide domain and narrow domain with respect to

categories, which compose this domain. The problem of classification of narrow domain corpora was considered in detail in the dissertation [Pinto, 2008].

In the present paper we deal with a relatively narrow domain: the intersection of lexis between some categories is 16%-25%. The results of measurements are presented in Table 5.1.

Table 5.1 Intersection of lexis

	Categories 1-2	***Categories 2-3***	***Categories 3-4***	***Categories 4-5***	***Categories 5-6***
Common words	23%	23%	16%	25%	19%

The experiments show that RUBRYX can easily cope with the difficulties caused by the mentioned intersection of lexis. The results we obtained are enough good and they exceed those obtained early in [Catena, 2008].

Conclusions

In the paper we tested the document classifier RUBRYX on a limited set of primary medical records. This set can be considered as a relatively narrow domain collection. We studied the sensitivity of classification results to threshold variations, use of terminological vocabulary and size of training set.

The experiments show that

- RUBRYX is easy tuned automatically and manually on a given corpus that allows to reach high results
- one word terms, bigrams and trigrams taken together and also taking into account their mutual position in a document allow to process narrow domain collections

In future we plan to combine the pre-processing procedure of RUBRYX with other classifiers such as Naïve Bayes, SVM, etc.

Acknowledgement

The authors are very appreciated to Mr. Vladimir Sinitsyn, one of RUBRYX developers, for his numerous consultations and additional software tools he offered for our work

Bibliography

[Aronow, 1995a] D.B.Aronow, J.R.Cooley, S.Soderland. Automated identification of episodes of asthma exacerbation for quality measurement in a computer-based medical record. In: Proc. of Annual Symposium on Computer Applications in Medical Care. 309-13, USA, 1995.

[Aronow, 1995b] D.B.Aronow, et.al. Automated classification of encounter notes in a computer based medical record. In: Proc. of MEDINFO '95 8th World Congress on Medical Informatics, Medinfo, Canada, p. 8-12, 1995.

[Baeza, 1999] R. Baeza-Yates, B. Ribeiro-Neta. Modern Information Retrieval. Addison Wesley, 1999

[Bishop, 2006] C. Bishop. Pattern Recognition and Machine Learning, Springer, 2006

[Catena, 2008] A.Catena, M.Alexandrov, B.Alexandrov, M.Demenkova. NLP-Tools Try To Make Medical Diagnosis. In: Proc. of the 1-st Intern. Workshop on Social Networking (SoNet-2008), Skalica, Slovakia, 2008.

[CLUTO, http] CLUTO: http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview

[Expressions, http] Expressions: http://en.wikipedia.org/wiki/Regular_expression

[Kaurova, 2011] E. Kaurova, M. Alexandrov, X. Blanco. Classification of free text clinical narratives (short review). In: Scient.Book “Information Science and Computing”, Publ. House ITHEA, 2011, 12 pp.

[Lopez, 2011] R.Lopez, M.Alexandrov, D,Barreda, J.Tejada. Proc. of the 4-th Intern. Conf.on Intelligent Information and and Engineering Systems (INFOS-2011), Publ. House ITHEA, Poland, 2011, 8 pp.

[Manning, 1999] C.D.Manning, H.Schutze, Foundations of statistical natural language processing. MIT Press, 1999

[Manning, 2009] C.D.Manning, H.Schutze, Introduction to Information Retrieval. Cambridge, 2009

[Mitchell, 1997] T.Mitchell. Machine Learning, McGrow Hill, 1997

[Pinto, 2008] D. Pinto, On Clustering and Evaluation of Narrow Domain Short-Text Corpora. Doctoral Dissertation, Polytechnic University of Valencia, Spain, 2008

[Polyakov, 2001] V. Polyakov, V. Sinitsyn. Method for automatic classification of web-resource by patterns in text processing and cognitive technologies. In: Text Collection, No.6, Publ. House Otechestvo, p. 120-126, 2001 (rus.)

[Polyakov, 2003] V.Polyakov, V. Sinitsyn. RUBRYX: technology of text classification using lexical meaning based approach. In: Proc. of Intern. Conf. Speech and Computing (SPECOM-2003), Moscow, MSLU, p. 137-143, 2003

[R, http] R-project: http:// www.r-project.org

[RText, http] R-project: http://cran.es.r-project.org/web/views/NaturalLanguageProcessing.html

[RapidMiner, http] RapidMiner: http://rapid-i.com

[Røst, 2006] T.B.Røst, O.Nytro, A.Grimsmo. Classifying Encouter Notes in the Primary Care Patient Record. In: Proc. of the 3-rd Intern. Workshop on Text-Based Information Retrieval (TIR-06). Univ. Press, p. 5-9, 2006.

[Rubryx, http] Rubryx: http://www.sowsoft.com/rubryx/rubryx2.zip

[Stein, 2003a] B.Stein,S.M.Eissen. AISearch: Category Formation of Web Search, www.aisearch.de, 2003

[Stein, 2003b] B.Stein, S.M.Eissen, F. Wissbrock. On Cluster Validity and the Information Need of Users. In: Proceedings of the 3rd IASTED International Conference on Artificial Intelligence and Applications (AIA'03), Benalmadena, Spain, Acta Press, pp.216-221, 2003

[Weka, http] Weka: http://www.cs.waikato.ac.nz/ml/weka/

[WekaText, http] Weka: http://stackoverflow.com/questions/7213125/building-running-a-streaming-weka-text-classifer-in-java

[Zhang, 2010] J.Zhang, Y.Gu, W.Liu, T.Zhao, X.Mu, W.Hu. Automatic Patient Search for Breast Cancer Clinical Trials Using Free-Text Medical Reports'. In: Proc. of the 1-st ACM International Health Informatics Symposium. USA., 2010.

Appendix

An example of primary medical record (in Spanish)

Quejas:dolores permanentes sordos en la zona iliaca.

Anamnesis de la enfermedad: El paciente se sintio enfermo hacia las 15 horas 10.02.07, cuando surgieron los dolores sordos vagos en el mesogastrio, nauseas, escalofrios. Despues de algunas horas los dolores se extendieron a la zona iliaca derecha. La ausencia de mejora hizo que llamara para pedir ayuda medica. Se traslado con sospecha de apendicitis aguda al hospital KB¹119 por el servicio de ambulancias.

Anamnesis de vida. Ha crecido y se ha desarrollado con normalidad. No hay enfermedades heredadas.

Enfermedades sufridas: No habia traumas u operaciones. Rechaza la anamnesis ulcerosa y cardial.

Anamnesis de alergia: No es relevante.

Diagnostico objetivo: estado de gravedad media. La epidermis es de color normal, humeda. La temperatura del cuerpo es de 37.2 C. La hemodinamica es estable. Pulso - 84. Presion arterial – 130/80 mm. La respiracion se realiza llenando por completo los pulmones. Frecuencia respiratoria 16 por minuto. La lengua es seca, con la placa blanca. El vientre es suave, indoloro en la zona iliaca derecha. Los sintomas de Rovzing, Sytkovskiy, Bartomie-Mihelson son positivos. No hay sintomatologia de peritonitis. La palpacion en la zona de cintura no produce dolor. No hay dificultades en la evacuacion urinaria y defecacion.

Se prescribe hospitalizacion en el area quirurgica e intervencion quirurgica urgente.

Authors' Information

Olga Kaurova *– Saint Petersburg State University (Department of Theoretical and Applied Linguistics - graduated in 2009); Autonomous University of Barcelona (International Master in "Natural Language Processing & Human Language Technology" - graduated in 2010; PhD program "Lenguas y Culturas Románicas" - current), 08193 Bellaterra (Barcelona), Spain;*

e-mail: kaurovskiy@gmail.com

Major Fields of Scientific Research: document classification, sentiment analysis

Mikhail Alexandrov *– Professor of the Academy of national economy and civil service under the President of Russia; Prosp. Vernadskogo 82, bld. 1, Moscow, 119571, Russia; research fellow of the fLexSem Research Group, Autonomous University of Barcelona, 08193 Bellaterra (Barcelona), Spain;*

e-mail: MAlexandrov@mail.ru

Major Fields of Scientific Research: data mining, text mining, mathematical modelling

Ales Bourek *– Senior lecturer, Masaryk University, Brno, Czech Republic; Head of Center for Healthcare Quality, Masaryk University. Kamenice 126/3, 62500 Brno, CZ. e-mail: bourek@med.muni.cz*

Major Fields of Interest: reproductive medicine – gynecology, health informatics, healthcare quality improvement, health systems

BUILDING THE LIBRARY CATALOG SEARCH MODEL BASED ON THE FUZZY SIMILARITY RELATION

Liliya Vershinina, Mikhail Vershinin, Andrej Masevich

Abstract: *We describe our approach to building the model of the search in libraries' catalogues based on the fuzzy similarity relation. To construct the model we carried out an experimental search to the variants of name in the catalogs of two libraries - the German National Library and the National Library of France. The model we constructed is based on the result of our experiment.*

Keywords*: fuzzy similarity relation, names transliteration, authority control, library catalogs*

ACM Classification Keywords*: H.3.6 Library Automation H.3.3, Information Search and Retrieval, 1.5.1 Pattern recognition. Fuzzy set*

Introduction

The implementation of fuzzy logic in the electronic catalogs is one of the requirements for the software of current library information systems.

Taking as examples the catalogues of two national libraries, we described from the users' point of view a search tool, which is embedded into the search system of these catalogues. It ensures taking into account during the search the spelling versions of the search terms and the elements of the records. The tool is likely based on the fuzzy sets theory.

Using the fuzzy sets theory, we built a general mathematical model on the base of which an instrument of this kind could be constructed.

For our research we used the catalogues of the German National Library (Deutsche Nationalbibliothek), of the National Library of France (Bibliothèque nationale de France), of the search portal of the European Library and of the Library of Congress Authorities. We selected just one type of element variation – the differences in the spellings of the transliteration of the name of the Russian composer Piotr Chajkovskij in the Latin alphabet. This name was chosen because it has many various forms in Latin scripts.

Transliteration of Cyrillic Script in the Latin alphabet

The transliteration of Russian texts in Latin characters has a long history and various traditions both in Russia and in other countries, where different languages and different systems of writing are used [Reformatskij A.A.1972].

The selection of the version of transliteration depends on many factors: in the first place, apparently, on the phonetic systems of the target languages. It is necessary to note that the application of various forms of the transliteration of names has a diachronic aspect, i.e. it changes with time.

These changes are well outlined on the graphs constructed with the aid of the Google n – gram viewer (http://books.google.com/ngrams), which makes it possible to determine the frequency of the occurrence of the form of words in texts of several millions of books in different languages and to build the graphs of the changes of this value with time.

For graphing, we selected five most common versions of transliteration (1.Tchaikovsky, 2.Chaikovsky, 3.Cajkovskij, 4.Čajkovskij, 5.Tchaikowsky). At least four regularities are distinctly visible on the graphs (Fig.1-3).

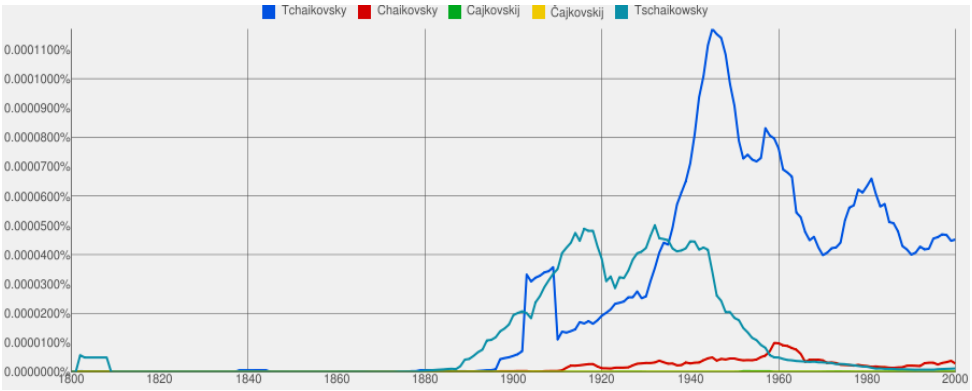


Fig.1 The frequency of the diverse variants of writing Chajkovskij in English 1800 -2000

First, in each of the three languages more than one version of the transliteration is present.

Secondly, different versions predominate in the different languages.

Thirdly, the frequency of the occurrence of each version changes in the course of time. The appearance of new versions is noted.

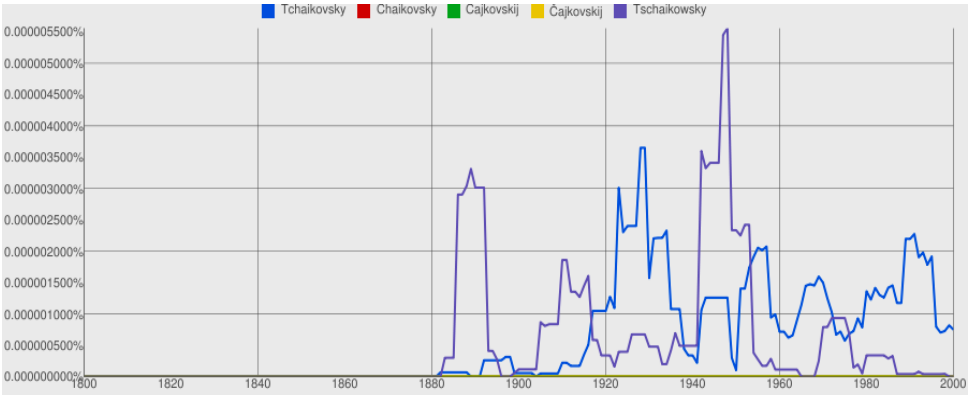


Fig.2 The frequency of the diverse variants of writing Chajkovskij in French 1800 -2000

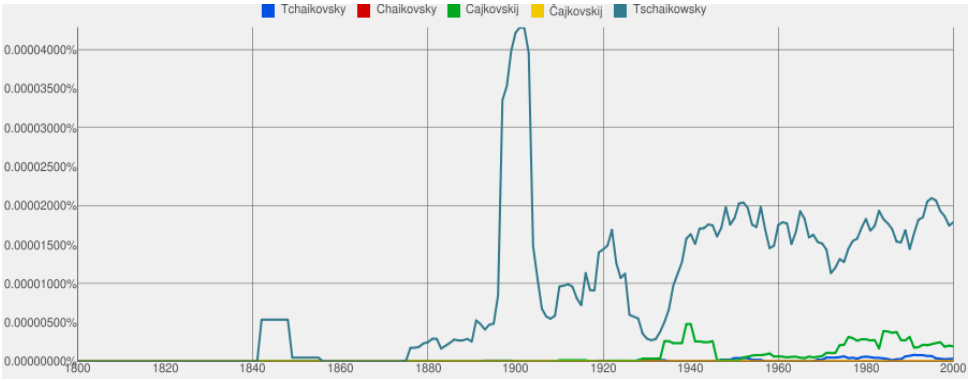


Fig.3 The frequency of the diverse variants of writing Chajkovskij in German 1800 -2000

Fourthly, in English and French in the different periods, versions 1 and 5 predominate, and in German, the steady predominance of version 5 is noted.

Currently in Russia the state standard GOST 7.79-2000 “Rules of transliteration of Cyrillic script in Latin alphabet” is accepted; it is the Russian version of the international standard ISO 9.95.

The standard proposes two versions of the transliteration – the strict one , where one character of the source alphabet is substituted by only one character of the target alphabet (system A, table 2 for non-Slavic languages) and the weakened one, where one character of the source language can be represented by more than one character of the target language [GOST 7.70-2000, 2001].

Thus, according to the standard, the surname “Chajkovskij” when transliterated in the Latin alphabet can appear in two versions:

Čajkovskij System A

Chajkovskij System B

Let us note that version 4, transliterated according to System A, which, therefore, corresponds to the international standard ISO 9.95, is encountered in none of the three natural languages (Fig. 1-3). We can easily explain this – the character Č is not used in these languages.

Records on the personal name "Chajkovskij" in the authority file of three national libraries

We compared authority records for the personal name Tschaikovsky P.I. from three sources: the authority files of the Library of Congress, of the national library of France and of the German national library.

The accepted transliteration form in the Library of Congress Authorities is Tchaikovsky, Peter Ilich, 1840-1893 (http://lccn.loc.gov/n79072979). This form, as we can see, does not conform to the standard ISO 9.95. The authority record presents 51 versions of the name spelling. However, it includes versions in Russian, Hebrew, Arabic and Chinese characters. The number of versions in Latin is thus 47.

The authority file of the German National Library (Gemeinsame Normdatei (GND)) accepted as heading the form Čajkovskij, Pëtr I. (http://d-nb.info/gnd/118638157), which fully conforms to the ISO standard; the record has 85 versions as references only in the Roman alphabet.

In the authority file of the national library of France (Autorités BnF) the chosen heading is Čajkovskij, Petr Il'ič (1840-1893) (http://catalogue.bnf.fr/ark:/12148/cb13900329p/PUBLIC). The transliteration conforms to ISO 9.95 with one exception: according to the standard the Russian character «ë» should be transliterated as «ë». The record contains 9 references.

Results of our experimental search in the catalogs of the German National Library and the National Library of France

From the authority files of three libraries, we selected ten versions from which 3 records are versions of the full name and 7 records only of the surname. Then we performed a search using each version as a search term in the catalogues of the libraries and the union catalogues which are accessible on Protocol Z39.50 via the portal of the European Library (TEL) (http://theeuropeanlibrary.org). The result is given in Table1.

Table 1. The result of the search through the European Library portal

The form of the name	Source (Authority file of the library)	Number of retrieved records	The codes of libraries where at least one record with this term was retrieved	Number of libraries where at least one record with this term was retrieved
Tchaikovsky	The National Library of France	16496	AL, AT, BE, CH, CY, CZ, **DE**, DK, EL, ES, FI, **FR**, IE, IS, IT, LU, LV, NL, NO, RS, SI, SK, TR, UK	24
Cajkovskij	The National Library of France	10799	AL, AT, BA, BE, CH, **DE**, DK, ES, FI, **FR**, HR, HU, IS, IT, LI, LU, LV, NL, NO, RS, SE, SI, UK	23
Tchaïkovsky	The National Library of France	20129	AT, BA, BE, CH, **DE**, DK, ES, FI, **FR**, HR, HU, IS, IT, LI, LU, LV, NL, NO, RS, SE, SI, TR, UK	23
Tschaikowsky	German National Library	12941	CH, CY, CZ, **DE**, DK, EL, ES, FI, **FR**, IE, IS, IT, LU, LV, NL, NO, RS, SI, TR, UK	20
Tchaikovsky, Peter Ilich	Library of Congress	10450	CH, CY, CZ, **DE**, DK, EL, ES, FI, **FR**, IE, IS, IT, LU, LV, NL, NO, RS, SI, UK	19
Čajkovskij	The National Library of France	11125	BE, CZ, DK ES, **FR**, HU, HR, IS, IT, LU, LV, NL, NO, RS, RU, SI, SK.TR, UK	17
Tchaikovsky, Pyotr Ilyich	Library of Congress	10262	BE, CZ, **DE**, ES, **FR**, HU, IS, IT, LU, LV, NL, NO, RS, RU, SI, SK.TR, UK	17
Chaikovsky	German National Library	7693	AT, CZ, **DE**, ES, FI, **FR**, IS, IT, LV, RS, RU, TR, UK	13
Tchaĭkovsky, Piotr Ilitch	Library of Congress	5	ES, IT	2

It is evident from Table 1 that the positive result of the search (at least one record with the term was retrieved) is obtained in the average in 16 libraries. In many libraries several forms are found. We selected two libraries.

German National Library (DNB) and National Library of France (BnF). In each of them a positive result was obtained with the search according to 8 versions. In the catalogues

of these two libraries, the search according to several versions of the name was carried out.

We presumed that if the difference between numbers of retrieved records was not significant, the catalogue had the program tool, based possibly on the fuzzy set logic, which takes in account a certain number of spelling versions. The search results are given in tables 2-3

Table 2. Results of our search according to the variants of the name in the catalog of the German national library

The spelling form of the name	Number of retrieved records
Tchaikovsky, Peter Ilich	6960
Tchaikovsky, Pyotr Ilyich	7122
Cajkovskij	7214
Tchaïkovsky	8503
Čajkovskij	7214
Tchaikovsky	8503
Tschaikowsky	10290
Chaikovsky	6991
Average number of record per a version	7849,63
Maximal (Tschaikowsky)	10290
Minimal (Tchaikovsky, Peter Ilich)	6960

It is evident from Table 2 that the numbers of records retrieved for each version are close. Differences in the number of obtained records with the search only on the surname can be caused by the presence in the catalog of the namesakes of the composer

It is evident from Table 3 that the results of our search in the catalogue of the National Library of France are different from the results in the German Library. Difference between the results of searches on various forms of the complete name between the numbers of records does not exceed 25. In five queries out of six the complete form of the name is used and, evidently, there is no factor of homonymy, i.e., the namesakes.

Table 3. Results of our search according to the variants of the name in the catalog of the national library of France

The spelling form of the name	Number of retrieved records
Tchaikovsky, Piotr Illitch	5360
Tchaïkowsky, Piotr Illitch	5365
Czajkowski, Piotr	5361
Tchaĭkovsky, Piotr Ilitch	5386
Čajkovskij, Petr Il'ič	5383
Tschaikowsky	424
Average number of record per a version	4546,5
Maximal (Tchaĭkovsky, Piotr Ilitch)	5386
Minimal (Tschaikowsky)	424

In the search using the version of the transliteration “Tschaikowsky”, a considerably smaller number of records was retrieved, which is apparently connected with the fact that this form is little used in French (see fig.3), whereas in German it has been most commonly used since 1920.

Data of our experimental search confirm, thus, the presence in the catalogues of these two libraries of the tool, which ensures the relative independence of the results of the search from the version of writing. Note that our research is a pilot one and our data have to be verified.

Some theoretical developments of the search in the library catalogue with the use of the fuzzy sets theory

In the process of designing, modifying and maintaining large libraries catalogues programmers, librarians and users deal with tasks in which one is supposed to operate with uncertain concepts and knowledge. The challenges of this kind are automation of documents indexing, multilingual search and the search by a term with more than one spelling.

To construct a relevant algorithm (of classification or record retrieval) it is necessary to formalize these concepts and knowledge. To carry out the description of uncertain concepts, and to operate many-valued incompletely specified lexical units and finally to build the models of the search, corresponding to the users’ information demand, one should apply the fuzzy sets theory.

The advantage of the fuzzy sets theory approach might be that within the framework of many-valued logic one can find solutions for a broader class of problems, than using clear logic.

We thus proposed [Vershinin M.I, 2000] to use a thesaurus with fuzzy relations between its elements (fuzzy thesaurus) for decreasing the expenditures for maintaining the catalogue and increasing the effectiveness in its use.

Further, we proposed [Vershinin M.I. et al, 2007] the algorithm of the automatic classification (indexing) of bibliographical records. The algorithm is based on the idea of the automatic fuzzy classification (indexing) of the records

The system attributes each bibliographic record to a definite class of terms marked with a subject heading. The assignment to the record of a subject heading index is determined by the comparison of the vocabulary of the record with the existing cluster of keywords, in this case the relation of similarity is uncertain. The classes do not have clear boundaries. That is why it cannot be determined unambiguously whether a record belongs to one class or another.

In the fuzzy classification, each document can be assigned to several classes with different degrees of membership. Several documents will be assigned to the same class if the degree of membership of the subject of each document to this class is maximal in comparison to the degrees of belonging to another class.

Thus, the possibility appears to perform a search by the uncertain attributes.

To deal with the problem of search and correction of errors of the system, a method of strings comparison based on the theory of fuzzy sets was developed [Vershinin M.I, 2001] The methodology of fuzzy logic permits to work under conditions when statistical data are lacking and to compare strings taking into account the possibility of errors without correcting the strings or without participation of an operator. The developed method takes in consideration both types of errors and their ranking according to the frequency of their occurrence and to other criteria.

Theoretical approach to building a search model based on the fuzzy sets theory

The formalization of uncertain concepts and relations is ensured by the introduction of linguistic and fuzzy variables, fuzzy set and fuzzy relation.

The fuzzy relations play in the theory of fuzzy sets and fuzzy logic an important role. Traditionally one applies fuzzy relations for modeling the structure of a complicated system, technological processes control and analysis of decision-making processes.

As far as maintaining the library catalogues is concerned, there is almost no practice of the use of fuzzy relations in that field. Below we shall try to show a perspective outlook of appliance of the method in order to ensure an effective search in the library catalogues.

The fuzzy relations theory is used as quality test for defining interrelations between the objects of the investigated system. Therewith the differences in the constraint force between objects are considered.

Commonly fuzzy n-ary relation is defined as a subset of Cartesian product of n sets [Pospelov D.A., ed., 1986]

$$R \subseteq X_1 \times X_2 \times \ldots \times X_n \tag{1}$$

and is specified with the membership function

$$\mu_R : \quad X_1 \times X_2 \times \ldots \times X_n \to L \tag{2}$$

As L one can take, for example, a set of real numbers, a segment of a real straight line, a set of linguistic variables, a set of m-dimensional vectors, pseudo Boolean algebra, completely distributive lattice etc.

This approach in the definition of L makes it possible to create various generalizations of the concept of relation, which one can apply to different fields. In addition, it allows using the well-developed set of devices of the relation theory, which results from the interpretation of a different function with the values from L.

As to the search in the library catalogues, the appliance of fuzzy relations gives the possibility to consider from a single point of view a variety of factors influencing the search quality, in particular to determine the links between query and catalogue records taking into account many factors.

To show the possibilities of solving several problems of search in library catalogues we confine ourselves to the consideration of binary fuzzy relations.

Generally one call fuzzy a binary relation between sets X and Y the function

$$R : \quad X \times Y \to L \tag{3}$$

Where L-is a completely distributive lattice, i.e. a partially ordered set, in which any non vacuous set has the greatest lower bound and least upper bound, and the operations of conjunction $\wedge$ and $\vee$ disjunction in L follow the distributive low. All the operations with fuzzy relations will be defined by these operations from L.

If we take a limited set of real numbers, then the operations of taking of greatest lower bound and the least upper bound will be correspondingly operations inf and sup, and the operations conjunction $\wedge$ and disjunction $\vee$ will be operations min and max. These operations will define also an operation with fuzzy relations.

In the case when L is a segment of a real straight line [0, 1], function R will be written as membership function

$$\mu_R : \quad X \times Y \to [0,1] \tag{4}$$

If the sets X and Y are bounded then the fuzzy relation between X and Y can be represented by its relation matrix, in which sets elements of X and Y correspond to lines and columns, and in the crossing of line x and column y the element R(x; y) is located.

In the case when sets X and Y coincide, the fuzzy relation R is called the fuzzy relation on the set X. To this relation one can assign a weighted graph in which each pair of knots (x;y) from X is connected with an arrow with weight R(x;y)

The model of search in the library catalogue based on the fuzzy similarity relation

Assuming that for the search in a catalogue the name of the Russian composer Chajkovskij Pyotr Il'ich is selected, the record for this name from the authority file of a certain library could be represented as a set of versions of its transliteration in Latin alphabet, which we designate by X.

Then the possible elements of set X are as follows:

x_1 – Tchaikovsky,

x_2 – Tchaïkovsky,

x_3 –Čajkovskij,

x_4 – Tschaikowsky,

x_5 – Chaikovsky etc.

Let us build a fuzzy similarity relation between versions taking in consideration diverse factors

(1) Account versions of transliteration (factor 1)

We present a fuzzy similarity relation as similarity matrix

$$M_1 = \{\mu_1(x_i; x_j)\}, i, j = 1,\ldots,m, \quad (5)$$

where $\mu_1(x_i; x_j)$ is the evaluation of similarity of the variants $\mu_1(x_i; x_j) \in [0,1]$.

The matrix of similarity can be obtained either by a quantitative evaluation of the certain parameter indicating the link between versions (number coincided characters, their sequence etc), or by questioning experts, who will indicate the degree of similarity for each pair of version from X in a certain scale of comparison which possibly consists of phrases like “very strong similarity”, “strong similarity”, “middle strong similarity”, “weak similarity”. It is obvious that in generally matrix M will be unsymmetrical.

(2) Account of occurrences of different versions of transliterations in diverse languages (factor 2)

To consider this factor we used the data of the diachronic search of the name Chajkovskij (fig1 – fig.3)

On the base of the data we built the matrix

$$A = \{a(x_i; t_j)\}, i = 1,\ldots, m, j = 1,\ldots, n \quad (6)$$

Where $a(x_i; t_j)$ – evaluation of the degree of occurrences of the name transliteration version x_i in the year t_j

The fuzzy similarity relation version R_2 can be represented as matrix of similarity

$$M_2 = A \bullet A^T \quad (7)$$

Where A^T is a transpose of matrix.

Note that the matrix M_2 has dimension $m \times m$ and is a symmetrical one.

Let us assume that the number of factors, which we have to take into account, is equal to k. After we defined matrixes of fuzzy relations M_1, M_2, …, M_k we build a matrix of fuzzy similarity relation M which takes into account all factors:

$$M = M_1 \wedge M_2 \wedge \ldots \wedge M_k \quad (8)$$

Now we use the fuzzy relation we built for the search in a library catalogue by the query z.

The algorithm looks as follows:

The similarity degree between query term z and versions of the set X is established. As a result we have a vector

$$\mu_z = \{\mu_z(x_1), \mu_z(x_2), \ldots, \mu_z(x_m)\}, \quad (9)$$

Where $\mu_z(x_i)$ – is evaluation of similarity degree of query z and record x_i (i=1,2,…, m).

To make evaluation of the similarity degree we can use the algorithm of fuzzy comparison of strings [3]. Note that the algorithm works even if the query formulation has errors.

(2) From the set X we select version x_{i0}, which

$$\mu_я(x_{i0}) = \max_i \mu(x_i), \quad i = 1,2,..., m \tag{10}$$

In the matrix M fuzzy relation we select the string with mark i_0 which correspond to the version x_{i0} This string provides a way to rank all the versions of set X. Essentially we got a corrected on the base of fuzzy relation vector μ_z. The records will be retrieved starting from the one, which contains version x_{i0} according to result of the ranking.

In the search technique one can also introduce a threshold α on the force of fuzzy similarity relation R, for example α = 0,5, and thus fix a selection of meaningful records.

The search starts from the record, which contains the term with the maximal degree of similarity. Note that the matrix of similarity M is generally asymmetrical and therefore when starting the search from different version, we may obtain different results.

Conclusion

It is evident from Tables 2-3 that the number of retrieved records varies depending on the form of name used in the query. The same result can be obtained by using the algorithm described above. We do not affirm that in the search systems of the catalogues of two libraries the fuzzy logic is realized. Possibly, in them probabilistic or any other model of the search is used

In our paper, we described a model of search where the fuzzy relation of similarity (matrices of similarity) is applied. For building the model, we use the result of our experimental research. The suggested model might be efficient in the design of search system.

Bibliography

[Vershinin M.I, 2000] Vershinin M.I. Sozdanie nechetkogo tezaurusa dlya e'lektronnogo kataloga in: Informacionny`e resursy` bibliotek i ix kadrovoe obespechenie: Mater. Mezhd. nauch. – prakt. konf., 23-26 maya 2000 g. / Belarus. un - t kul`tury`. - Minsk, 2000. - S. 91-96.

[Vershinin M.I. et al, 2007] Vershinin M.I., Vershinina L.P. Primenenie nechetkoj logiki v gumanitarny`x issledovaniyax in: Bibliosfera, 2007, №4. S.43-47.

[Vershinin M.I., 2001] Vershinin M.I. E'lektronny`j katalog: problemy` i resheniya. SPb.: Professiya, 2009.-232 s.

[GOST 7.70-2000, 2001] GOST 7.70-2000 Pravila transliteracii kirillovskogo pis`ma latinskim alfavitom. – M.: Gosstandart, 2001- 20 s.

[Pospelov D.A., ed., 1986] Nechetkie mnozhestva v modelyax upravleniya i iskusstvennogo intellekta / Pod red. D.A.Pospelova. M.: Nauka, 1986. 312 s.

[Reformatskij A.A., 1972] Reformatskij A.A. O standartizacii transliteracii latinskimi bukvami russkix tekstov in: Nauchno-texnicheskaya informaciya – 1972. -№ 10 S.32-36 [Zakharov V.P. et al, 1996]

[Zakharov V.P. et al] Zakharov V.P., Masevich A.C., Pimenov E.N. Authority control as a linguistic support element of an automated library system in: International Cataloguing and Bibliographic Control - 1996. - Vol 25, N4. - p.84-86.

Authors' Information

Liliya Vershinina *– Head of Department of Information Science and Mathematics, Saint Petersburg State University of Culture and Arts, e-mail: zk-inf@yandex.ru*

Major Fields of Scientific Research: General theoretical information research

Mikhail Vershinin*– Associate Professor. Department of Mechanics, National University of Mineral Mining "Gorny'j" ; e-mail:* stephen@smtp.ru

Major Fields of Scientific Research: Software technologies, General theoretical information research

Andrej Masevich *– Assistant professor. Saint-Petersburg State University of Culture and Arts; e-mail: andmasev@mail.ru*

Major Fields of Scientific Research: Computational Linguistics, Library information systems

STORING RDF GRAPHS USING NL-ADDRESSING

Krassimira Ivanova, Vitalii Velychko, Krassimir Markov

***Abstract**: NL-addressing is a possibility to access information using natural language words as addresses of the information stored in the multi-dimensional numbered information spaces. For this purpose the internal encoding of the letters is used to generate corresponded co-ordinates. The tool for working in such style is named OntoArM. Its main principles, functions and using for storing RDF graphs are outlined in this paper.*

***Keywords**: NL-addressing, RDF graphs, ontology representations.*

***ACM Classification Keywords**: D.4.2 Storage Management; E.2 Data Storage Representations.*

Introduction

Resource Description Framework (RDF) is the W3C recommendation for semantic annotations in the Semantic Web. RDF is a standard syntax for Semantic Web annotations and languages [Klyne & Carroll, 2004].

The underlying structure of any expression in RDF is a collection of triples, each consisting of a ***subject***, a ***predicate*** and an ***object***. A set of such triples is called an ***RDF graph***. This can be illustrated by a node and directed-arc diagram, in which each triple is represented as a node-arc-node link (hence the term "graph") (Fig.1).

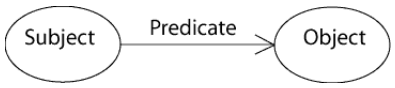


Fig. 1. RDF triple

Each triple represents a statement of a relationship between the things denoted by the nodes that it links. Each triple has three parts: (1) subject, (2) object, and (3) a predicate (also called a *property*) that denotes a relationship. The direction of the arc is significant: it always points toward the object. The nodes of an RDF graph are its subjects and objects.

The assertion of an RDF triple says that some relationship, indicated by the predicate, holds between the things denoted by subject and object of the triple. The assertion of an

RDF graph amounts to asserting all the triples in it, so the meaning of an RDF graph is the conjunction (logical AND) of the statements corresponding to all the triples it contains. A formal account of the meaning of RDF graphs is given in [Hayes, 2004].

The state of the art with respect to existing storage and retrieval technologies for RDF data is given in [Hertel et al, 2009]. Different repositories are imaginable, e.g. main memory, files or databases. RDF schemas and instances can be efficiently accessed and manipulated in main memory. For persistent storage the data can be serialized to files, but for large amounts of data the use of a database management system is more reasonable. Examining currently existing RDF stores we found that they are using relational and object-relational database management systems. Storing RDF data in a relational database requires an appropriate table design. There are different approaches that can be classified in (1) generic schemas, i.e. schemas that do not depend on the ontology, and (2) ontology specific schemas.

In the following we will present a new approach for organizing graph data bases, called Natural Language Addressing (NL-Addressing) and will illustrate it for the most important ontological table designs.

Natural Language Addressing (NL-Addressing)

The idea of Natural Language Addressing (NL-Addressing) is very simple. It is based on the computer internal representation of the word as strings of codes in any system of encoding (ASCII, UNICODE, etc.).

For example, the ASCII encoding of the word „accession" has the next representation: 97 99 99 101 115 115 105 111 110. It may be used as co-ordinate array, which indicates a point in the multidimensional information space, where the corresponded information may be stored.

It is clear, the words have different lengths and, in addition, some phrases may be assumed as single concepts. This means that we need a tool for managing multidimensional information spaces with possibility to support all needed dimensions in one integrated structure.

The independence of dimensionality limitations is very important for developing new intelligent systems aimed to process high-dimensional data. To achieve this, we need information models and corresponding access methods to cross the boundary of the dimensional limitations and to obtain the possibility to work with information spaces with variable and practically unlimited number of dimensions. Such possibility is given by the Multi-Dimensional Information Model (MDIM) [Markov, 2004] and correspond Multi-Dimensional Access Method (MDAM) [Markov, 1984]. Its advantages have been

demonstrated in many practical realizations during more than twenty-five years. In recent years, this kind of memory organization has been implemented in the area of intelligent systems memory structuring for several data mining tasks and especially in the area of association rules mining [Mitov et al, 2009]. Here we will show its applicability for organizing of RDF stores.

Multi-dimensional numbered information spaces

Main structures of Multi-Dimensional Information Model (MDIM) are *basic information elements, information spaces, indexes* and *meta-indexes,* and *aggregates*. The definitions of these structures are remembered below.

The ***basic information element*** (***BIE***) of MDIM is an arbitrary long string of machine codes (bytes). When it is necessary, the string may be parceled out by lines. The length of the lines may be variable.

Let ***the universal set UBIE*** be the set of all ***BIE***.

Let $\boldsymbol{E}_1$ be a set of basic information elements. Let μ_1 be a function, which defines a biunique correspondence between elements of the set $\boldsymbol{E}_1$ and elements of the set $\boldsymbol{C}_1$ of positive integer numbers, i.e.:

$$\boldsymbol{E}_1 = \{e_i \mid e_i \in \boldsymbol{UBIE}\ , i=1,\ldots, m_1\},\ \boldsymbol{C}_1 = \{c_1 \mid c_i \in \boldsymbol{N}, i=1,\ldots, m_1\};\quad \boldsymbol{\mu_1 : E_1 \leftrightarrow C_1}$$

The elements of $\boldsymbol{C}_1$ are said to be numbers (co-ordinates) of the elements of $\boldsymbol{E}_1$.

The triple $\boldsymbol{S_1 = (E_1, \mu_1, C_1)}$ is said to be a ***numbered information space of range 1*** (one-dimensional or one-domain information space).

Let $\boldsymbol{NIS}_1$ be a set of all one-dimensional information spaces.

The triple $\boldsymbol{S_2 = (E_2, \mu_2, C_2)}$ is said to be a ***numbered information space of range 2*** (two-dimensional or multi-domain information space of range two) iff the elements of $\boldsymbol{E}_2$ are numbered information spaces of range one (i.e. belong to the set $\boldsymbol{NIS}_1$) and μ_2 is a function which defines a biunique correspondence between elements of $\boldsymbol{E}_2$ and elements of the set $\boldsymbol{C}_2$ of positive integer numbers, i.e.:

$$\boldsymbol{E}_2 = \{e_i \mid e_i \in \boldsymbol{NIS}_1\ , i=1,\ldots, m_2\},\ \boldsymbol{C}_2 = \{c_i \mid c_i \in \boldsymbol{N}, i=1,\ldots, m_2\};\quad \boldsymbol{\mu_2 : E_2 \leftrightarrow C_2}$$

Let $\boldsymbol{NIS}_{n-1}$ be a set of all (n-1)-dimensional information spaces.

The triple $\boldsymbol{S_n = (E_n, \mu_n, C_n)}$ is said to be a ***numbered information space of range n*** (n- dimensional or multi-domain information space) iff the elements of $\boldsymbol{E}_n$ are numbered information spaces of range n-1 (belong to the set $\boldsymbol{NIS}_{n-1}$) and μ_n is a function which

defines a biunique correspondence between elements of $\boldsymbol{E}_n$ and elements of the set $\boldsymbol{C}_n$ of positive integer numbers, i.e.:

$\boldsymbol{E}_n = \{e_j \mid e_j \in \boldsymbol{NIS}_{n-1}, j=1,\ldots, m_n\}$, $\boldsymbol{C}_n = \{c_j \mid c_j \in \boldsymbol{N}, j=1,\ldots, m_n\}$; $\boldsymbol{\mu_n : E_n \leftrightarrow C_n}$

The information space $\boldsymbol{S}_n$, which contains all information spaces of a given application is called ***information base*** of range **n**. The concept information base without indication of the range is used as generalized concept to denote all available information spaces.

The sequence $\boldsymbol{A} = (c_n, c_{n-1}, \ldots, c_1)$, where $c_i \in \boldsymbol{C}_i$, $i=1, \ldots, n$ is called ***multidimensional space address*** of range ***n*** of a basic information element. Every space address of range ***m***, ***m*** **<** ***n***, may be extended to space address of range ***n*** by adding leading *n-m* zero codes. Every sequence of space addresses $\boldsymbol{A_1, A_2, \ldots, A_k}$, where ***k*** is arbitrary positive number, is said to be a ***space index***.

Every index may be considered as a basic information element, i.e. as a string, and may be stored in a point of any information space. In such case, it will have a multidimensional space address, which may be pointed in the other indexes, and, this way, we may build a hierarchy of indexes. Therefore, every index, which points only to indexes, is called ***meta-index***.

The approach of representing the interconnections between elements of the information spaces using (hierarchies) of meta-indexes is called ***poly-indexation*****.**

Let $G = \{S_i \mid i=1, \ldots, n\}$ be a set of numbered information spaces.

Let $\boldsymbol{\tau} = \{\nu_{ij} : S_i \rightarrow S_j \mid i=\text{const}, j=1, \ldots, n\}$ be a set of mappings of one "main" numbered information space $\boldsymbol{S}_i \in G \mid i=\text{const}$, into the others $\boldsymbol{S}_J \in G$, $j=1, \ldots, n$, and, in particular, into itself.

The couple: $D = (G, \boldsymbol{\tau})$ is said to be an "***aggregate***".

It is clear, we can build **m** aggregates using the set G because every information space $\boldsymbol{S}_J \in G$, $j=1, \ldots, n$, may be chosen to be the main information space.

Operations in the MDIM

After presenting the information structures, we need to remember the operations, which are admissible in the model. In MDIM, we assume that ***all*** *information elements of* ***all*** *information spaces* ***exist***.

If for any $\boldsymbol{S}_i$: $\boldsymbol{E}_i = \emptyset \wedge \boldsymbol{C}_i = \emptyset$, than it is called ***empty***.

Usually, most of the information elements and spaces are empty. This is very important for practical realizations.

Because of the rule that all structures exist, we need only two operations with a ***BIE***: updating and getting the value and two service operations: getting the length of a ***BIE*** and positioning in a ***BIE***.

Updating, or simply – ***writing*** the element, has several modifications with obvious meaning: writing as a whole; appending/inserting; cutting/replacing a part; deleting.

There is only one operation for getting the value of a ***BIE***, i.e. ***read*** a portion from a ***BIE*** starting from given position. We may receive the whole ***BIE*** if the starting position is the beginning of ***BIE*** and the length of the portion is equal to the ***BIE*** length.

We have only one operation with a **single space** – *clearing (deleting) the space*, i.e. replacing all ***BIE*** of the space with Ø (empty ***BIE***). After this operation, all ***BIE*** of the space will have zero length. Really, the space is cleared via replacing it with empty space.

We may provide two operations with **two spaces**: (1) *copying* and (2) *moving* the first space in the second. The modifications concern how the ***BIE*** in the recipient space are processed. We may have: copy/move with clearing the recipient space; copy/move with merging the spaces.

The first modifications first clear the recipient space and after that provide a copy or move operation. The second modifications may have two types of processing: destructive or constructive. The ***destructive merging*** may be "conservative" or "alternative". In the conservative approach, the ***BIE*** of recipient space remains in the result if it is with none zero length. In the other approach – the ***BIE*** from donor space remains in the result. In the ***constructive merging*** the result is any composition of the corresponding ***BIE*** of the two spaces.

Of course, the move operation deletes the donor space after the operation.

Special kind of operations concerns the *navigation* in a space. We may receive the space address of the ***next*** or ***previous***, ***empty*** or ***non-empty***, elements of the space starting from any given co-ordinates.

The possibility to count the number of non empty elements of a given space is useful for practical realizations.

Operations with indexes, meta-indexes, and aggregates in the MDIM are based on the classical logical operations – intersection, union, and supplement, but these operations are not so trivial. Because of the complexity of the structure of the information spaces, these operations have two different realizations.

Every information space is built by two sets: the set of co-ordinates and the set of information elements. Because of this, the operations with indexes, meta-indexes, and aggregates may be classified in two main types: (1) operations based only on

co-ordinates, regardless of the content of the structures; (2) operations, which take in account the content of the structures:

- The operations based only on the co-ordinates are aimed to support information processing of analytically given information structures. For instance, such structure is the table, which may be represented by an aggregate. Aggregates may be assumed as an extension of the relations in the sense of the model of Codd [Codd, 1970]. The relation may be represented by an aggregate if the aggregation mapping is one-one mapping. Therefore, the aggregate is a more universal structure than the relation and the operations with aggregates include those of relation theory. What is the new is that the mappings of aggregates may be not one-one mappings.
- In the second case, the existence and the content of non empty structures determine the operations, which can be grouped corresponding to the main information structures: elements, spaces, indexes, and meta-indexes. For instance, such operation is the **projection**, which is the analytically given space index of non-empty structures. The projection is given when some coordinates (in arbitrary positions) are fixed and the other coordinates vary for all possible values of coordinates, where non-empty elements exist. Some given values of coordinates may be omitted during processing.

Other operations are transferring from one structure to another, information search, sorting, making reports, generalization, clustering, classification, etc.

OntoArM

The program realization of MDIM is called Multi-Domain Access Method (MDAM). For a long period, it has been used as a basis for organization of various information bases. There exist several realizations of MDAM for different hardware and/or software platforms. The most resent one is the FOI Archive Manager – ArM [Markov et al, 2008]. The newest MDAM realization is called ArM32 (for MS Windows). [Markov, 2004]

The OntoArM is an ontological graph oriented access method but not a middleware in the sense of [Hertel et al, 2009]. It is an upgrade of ArM32.

The OntoArM ontological elements are organized in ontological graph spaces with variable ranges. There is no limit for the ranges of the spaces. Every ontological element may be accessed by a corresponding multidimensional space address (coordinates) given

via NL-word or phrase. Therefore, we have two main constructs of the physical organizations of OntoArM – ontological spaces and ontological elements.

In OntoArM the length of the ontological element (string) may vary from 0 up to 1G bytes. There is no limit for the number of strings in an archive but their total length plus internal indexes could not exceed the limited length of the file system for a single file (4G, 8G, etc.). There is no limit for the numbers of files in the information base as well as for theirs dispositions.

OntoArm operations inherited from ArM32

The operations with basic information elements are:

- *ArmRead* (reading a part or a whole element);
- *ArmWrite* (writing a part or a whole element);
- *ArmAppend* (appending a string to an element);
- *ArmInsert* (inserting a string into an element);
- *ArmCut* (removing a part of an element);
- *ArmReplace* (replacing a part of an element);
- *ArmDelete* (deleting an element);
- *ArmLength* (returns the length of the element in bytes).

The operations over the spaces are:

- *ArmDelSpace* (deleting the space),
- *ArmCopySpace* and *ArmMoveSpace* (copying/moving the first space in the second in the frame of one file),
- *ArmExportSpace* (copying one space from one file the other space, which is located in other file).

The operations, aimed to serve the navigation in the information spaces return the space address of the ***next*** or ***previous***, ***empty*** or ***non-empty*** elements of the space starting from any given co-ordinates. They are *ArmNextPresent, ArmPrevPresent, ArmNextEmpty*, and *ArmPrevEmpty*.

The projections' operations return the space address of the ***next*** or ***previous non-empty*** elements of the projection starting from any given co-ordinates. They are *ArmProjNext* and *ArmProjPrev*.

The operations, which create indexes, are:

- *ArmSpaceIndex* – returns the space index of the non-empty structures in the given information space;
- *ArmProjIndex* – gives the space index of basic information elements of a given projection

The service operations for counting non-empty elements or subspaces are correspondingly:

- *ArmSpaceCount* – returns the number of the non-empty structures in given information space;
- *ArmProjCount* – gives the number of elements of given (hierarchical or arbitrary) projection.

OntoArm RDF graph oriented operations

Converting strings into space addresses

There are two internal operations for conversion:

- *ArmStr2Addr* – converts string to space address. Four ASCII symbols or two UNICODE 16 symbols form one co-ordinate word. This reduces four, respectively – two, times the space' dimensions. The string is extended with leading zeroes if it is needed.
- *ArmAddr2Str* – converts space address in ASCII or UNICODE string. The leading zeroes are not included in the string.

The operations for conversion are not needed for the end-user because they are used by the upper level operations given below. All OntoArM operations access the information by NL-addresses (given by a NL-words or phrases). Because of this we will not point specially this feature.

OntoArM operations for storing and receiving RDF information

There are two main operations for creating the RDF-store:

- *OntoArmWrite* – writes a buffer (usually NL-string).
- *OntoArmRead* – reads a buffer (usually NL-string).

It is clear; to work easily with RDF graphs, several additional operations are needed:

- *OntoArmAppend (appending a string to an element);*
- *OntoArmInsert (inserting a string into an element);*
- *OntoArmCut (removing a part of an element);*
- *OntoArmReplace (replacing a part of an element);*
- *OntoArmDelete (deleting an element);*
- *OntoArmLength (returns the length of the element in bytes).*

OntoArM operations for graph navigation

The operations, aimed to serve the navigation in the graph are context depended – the format of the elements is important for the navigation. If the element is an NL-index, the navigation operation may take its ***next*** or ***previous*** NL-word for further processing. If the element has more complicated structure, the navigation operations have to be accommodated to it. In general, these operations are usual ones for navigating in the graph structures.

NL-Addressing for ontology generic schemas

Vertical representation

The simplest RDF generic schema is the triple store with only one table required in the database. The table contains three columns named *Subject*, *Predicate* and *Object*, thus reflecting the triple nature of RDF statements. This corresponds to the *vertical representation* for storing objects in a table [Agrawal et al, 2001].

The greatest advantage of this schema is that no restructuring is required if the ontology changes. Adding new classes and properties to the ontology can be realized by a simple INSERT command in the table. On the other hand, performing a query means searching the whole database and queries involving joins become very expensive. Another aspect is that the class hierarchy cannot be modeled in this schema, what makes queries for all instances of a class rather complex [Hertel et al, 2009].

It is easy to store this schema via OntoArM. The *Subject* will be the address and all its couples (*Predicate*, *Object*) may be stored at one and the same address. This way with one operation all arcs of the node of the graph will be received. There exists another variant of organization where the *Predicate* may be additional co-ordinate or name of the archive. In this case, additional operations for reading arcs will be needed. Nevertheless, in all cases the OntoArM will have linear complexity O(max_L), where max_L is the maximal length of the word or phrases, used for NL-addressing. In the same time, the relational table has complexity at least O(n log n), where n is number of all indexed elements (words), if we will take in account supporting indexing and binary search. Of course, the memory for binary indexes exceeds the OntoArM memory for internal indexes. At the end, the time for direct access is many times less then via binary search. The speed experiments with *Firebird* relation data base had showed about 30-ty times for reading and more than 90-ty times for writing in ArM's favor [Markov et al, 2008].

Normalized triple store

The triple store can be used in its pure form [Oldakowski et al, 2005], but most existing systems add several modifications to improve performance or maintainability. A common approach, the so-called *normalized triple store*, is adding two further tables to store resource URIs and literals separately as shown in Fig. 2, which requires significantly less storage space [Harris & Gibbins, 2003]. Furthermore, a hybrid of the simple and the normalized triple store can be used, allowing storing the values themselves either in the triple table or in the resources table [Jena2, 2012].

Trilpes:

Subject	Predicate	IsLiteral	Object
r1	*r2*	*False*	*r3*
r1	*r4*	*True*	*I1*
…	…	…	…

Resources:

ID	URI
r1	*…#1*
r2	*…#2*
…	…

Literals:

ID	Value
I1	*Value1*
…	…
…	…

Fig. 2. Normalized triple store

In a further refinement, the Triples table can be split horizontally into several tables, each modeling an RDF(S) property:

- SubConcept for the rdfs:subClassOf property, storing the class hierarchy
- SubProperty for the rdfs:subPropertyOf property, storing the property hierarchy
- PropertyDomain for the rdfs:domain property, storing the domains and cardinalities of properties
- PropertyRange for the rdfs:range property, storing the ranges of properties
- ConceptInstances for the rdf:type property, storing class instances
- PropertyInstances for the rdf:type property, storing property instances
- AttributeInstances for the rdf:type property, storing instances of properties with literal values

These tables only need two columns for *Subject* and *Object*. The table names implicitly contain the predicates. This schema separates the ontology schema from its instances, explicitly models class and property hierarchies and distinguishes between class-valued and literal-valued properties [Broekstra, 2005; Gabel et al, 2004].

The normalized triple store is ready for representing via OntoArM. Only what we have to do is to take in account the representing all arcs from a node by one space NL-index and the representing all properties as an aggregate. The *Subject* will be the NL-address and only *Object* will be saved. Possibility to concatenate all *Objects* for a *Subject* reduces the size of memory and time. There are different approaches for building the aggregate – using additional co-ordinate to the *Subjects'* values or to use separate archives for storing the information.

In all cases, the OntoArM has linear complexity O(max_L), the relation data base – at least O(n log n).

NL-Addressing for ontology specific schemas

Horizontal representation

Ontology specific schemas are changing when the ontology changes, i.e. when classes or properties are added or removed. The basic schema consists of one table with one column for the instance ID, one for the class name and one for each property in the ontology. Thus, one row in the table corresponds to one instance. This schema is corresponding to the *horizontal representation* [Agrawal et al, 2001] and obviously has several drawbacks: large number of columns, high sparsity, inability to handle multi-valued properties and the need to add columns to the table when adding new properties to the ontology, just to name a few.

Horizontally splitting this schema results in the so called one-table-per class schema - one table for each class in the ontology is created. A class table provides columns for all properties whose domain contains this class. This is tending to the classic entity-relationship-model in database design and benefits queries about all attributes and properties of an instance.

However, in this form the schema still lacks the ability to handle multi-valued properties, and properties that do not define an explicit domain must then be included in each table. Furthermore, adding new properties to the ontology again requires restructuring existing tables [Hertel et al, 2009].

The horizontal representation is an example of a set of aggregates in the sense of OntoArM. Storing every class in a separate archive gives possibility to add properties without restructuring existing tables because the aggregate may be described by a meta-index. Again, NL-addressing in OntoArM has linear complexity O(max_L), the relation data base representation – at least O(n log n).

Decomposition storage model

Another approach is vertically splitting the schema, what results in the one-table-per-property schema, also called the *decomposition storage model*.

In this schema one table for each property is created with only two columns for *Subject* and *Object*. RDF(S) properties are also stored in such tables, e.g. the table for rdf:type contains the relationships between instances and their classes.

This approach is reflecting the particular aspect of RDF that properties are not defined inside a class. However, complex queries considering many properties have to perform many joins, and queries for all instances of a class are similarly expensive as in the generic triple schema [Hertel et al, 2009].

In practice, a hybrid schema combining the table-per-class and table-per property schemas is used to benefit from the advantages of both of them. This schema contains one table for each class, only storing there a unique ID for the specific instance. This replaces the modeling of the rdf:type property. For all other properties tables are created as described in the table-per-property approach (Fig. 3) [Pan & Heflin, 2004]. Thus, changes to the ontology do not require changing existing tables, as adding a new class or property results in creating a new table in the database.

ClassA:

ID
…#1
…

Property1:

Subject	Object
…#1	…#3
…	…

ClassB:

ID
…#3
…

Fig. 3. Hybrid schema

A possible modification of this schema is separating the ontology from the instances. In this case, only instances are stored in the tables described above.

Information about the ontology schema is stored separately in four additional tables *Class*, *Property*, *SubClass* and *SubProperty* [Alexaki et al, 2001]. These tables can be further refined storing only the property ID in the Property table and the domain and range of the property in own tables Domain and Range [Broekstra, 2005]. This approach is similar to the refined generic schema, where the ontology is stored the same way and only the storage of instances is different.

To reduce the number of tables, single-valued properties with a literal as range can be stored in the class tables. Adding new attributes would then require changing existing tables. Another variation is to store all class instances in one table called Instances. This is especially useful for ontologies where there is a large number of classes with only few or no instances [Alexaki et al, 2001].

The decomposition storage model is memory and time consuming due to duplicating the information and generation of too much binary search indexes. It is very near to the OntoArM style and may be directly implemented using NL-addressing but this will be not efficient. NL-addressing permits new possibilities due to omitting of explicit given information – names as well as binary indexes. The feature tables may be replaced by NL-addressing access to corresponded points of the information space where all information about given *Subject* will exist. This way we will reduce the needed memory and time. At the end, let point again, that NL-addressing has linear complexity O(max_L) and the relation data base representation – at least O(n log n).

Conclusion

NL-addressing is a possibility to access information using natural language words as addresses of the information stored in the multi-dimensional numbered information spaces. For this purpose the internal encoding of the letters is used to generate corresponded co-ordinates. The tool for working in such style is named OntoArM. Its main principles, functions and using for storing RDF graph were outlined in this paper.

There are further issues not pointed above, which may require an extension of the triple-based schemas and thus are affecting the design of the database: (1) Storing multiple ontologies in one database; (2) Storing statements from multiple documents in one database.

Both points are concerning the aspect of provenance, which means keeping track of the source an RDF statement is coming from. When storing multiple ontologies in one database it should be considered that classes, and consequently the corresponding tables, can have the same name. Therefore, either the tables have to be named with a prefix referring to the source ontology [Pan & Heflin, 2004] or this reference is stored in an additional attribute for every statement. A similar situation arises for storing multiple documents in one database. Especially, when there are contradicting statements it is important to know the source of each statement. Again, an additional attribute denoting the source document helps solving the problem [Pan & Heflin, 2004].

The concept of named graphs [Caroll et al, 2004] is including both issues. The main idea is that each document or ontology is modeled as a graph with a distinct name, mostly a URI. This name is stored as an additional attribute, thus extending RDF statements from triples to so-called quads. For the database schemas described above this means adding a fourth column to the tables and potentially storing the names of all graphs in a further table.

All these problems can be solved by OntoArM, because a separated ontology may be represented in one single archive. In addition, the NL-addressing permits accessing the equal names in different ontologies without any additional indexing or using of pointers, identification and etc. Only the NL-words or phrases are enough to access all information in all existing ontologies (resp. graphs).

The linear complexity O(max_L) of NL-addressing is very important for realizing very large triple stores.

OntoArM is implemented in the Institute of Cybernetics V.M. Glushkov at the National Academy of Sciences of Ukraine, Kiev (IC NASU). It has been used for storing ontology information about multiple documents from own data bases as well as from different internet sources.

The further work is concerned to implementing OntoArM for storing multiple ontologies in the libraries of the "Instrumental Complex with Ontological Purpose", which is under developing in the IC NASU.

Acknowledgements

The paper is partially financed by the project ITHEA XXI of the Institute of Information Theories and Applications FOI ITHEA and the Consortium FOI Bulgaria (www.ithea.org, www.foibg.com).

Bibliography

[Agrawal et al, 2001] Agrawal R, Somani A, Xu Y Storage and querying of e-commerce data. In: Proceedings of the 27th Conference on Very Large Data Bases, VLDB 2001,Roma, Italy.

[Alexaki et al, 2001] Alexaki S, Christophides V, Karvounarakis G, Plexousakis D, Tolle K (2001) The ICS-FORTH RDFSuite: Managing voluminous RDF description bases. In: Proceedings of the 2nd International Workshop on the Semantic Web, Hongkong.

[Broekstra, 2005] Broekstra J. Storage, querying and inferencing for Semantic Web languages. PhD Thesis, Vrije Universiteit, Amsterdam (2005).

[Caroll et al, 2004] Caroll J, Bizer C, Hayes P, Stickler P (2004) Semantic Web publishing using named graphs. In: Proceedings of Workshop on Trust, Security, and Reputation on the SemanticWeb, at the 3rd International SemanticWeb Conference, ISWC 2004, Hiroshima, Japan.

[Codd, 1970] Codd, E.: A relation model of data for large shared data banks. Magazine Communications of the ACM, 13/6, 1970, pp.377 387.

[Gabel et al, 2004] Gabel T, Sure Y, Voelker J (2004) KAON – An overview. Insititute AIFB, University of Karlsruhe. http://kaon.semanticweb.org/main kaonOverview.pdf.

[Harris & Gibbins, 2003] Harris S, Gibbins N 3store: Efficient bulk RDF storage. In: Proceedings of the 1st International Workshop on Practical and Scalable Semantic Systems, PSSS 2003, Sanibel Island, FL, USA.

[Hayes, 2004] Patrick Hayes, Editor, *RDF Semantics*, W3C Recommendation, 10 February 2004, http://www.w3.org/TR/2004/REC-rdf-mt-20040210/ . Latest version available at http://www.w3.org/TR/rdf-mt/ .

[Hertel et al, 2009] Alice Hertel, Jeen Broekstra, and Heiner Stuckenschmidt. RDF Storage and Retrieval Systems. In: S. Staab and R. Studer (eds.), Handbook on Ontologies, International Handbooks on Information Systems, DOI 10.1007/978-3-540-92673-3, Springer-Verlag Berlin Heidelberg 2009. pp 489-508.

[Jena2, 2012] Jena2 database interface – database layout. http://jena.sourceforge.net/DB/layout.html. (visited at 22.08.2012)

[Klyne & Carroll, 2004] Graham Klyne and Jeremy J. Carroll, Editors, *Resource Description Framework (RDF): Concepts and Abstract Syntax*, W3C Recommendation, 10 February 2004, http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/ . Latest version available at http://www.w3.org/TR/rdf-concepts/ .

[Markov et al, 2008] Markov K, Ivanova, K., Mitov, I., & Karastanev, S. Advance of the access methods. Int. J. Information Technologies and Knowledge, 2/2, 2008, pp.123-135

[Markov, 1984] Kr.Markov. A Multi-domain Access Method. // Proceedings of the International Conference on Computer Based Scientific Research. Plovdiv, 1984. pp. 558-563.

[Markov, 2004] Markov, K. Multi-domain information model. Int. J. Information Theories and Applications, 11/4, 2004, pp.303-308.

[Mitov et al, 2009] Mitov, I., Ivanova, K., Markov, K., Velychko, V., Vanhoof. K., Stanchev, P. "PaGaNe" – A classification machine learning system based on the multidimensional numbered information spaces. In World Scientific Proc. Series on Computer Engineering and Information Science, No.2, pp.279 286.

[Oldakowski et al, 2005] Oldakowski R, Bizer C, Westphal D RAP: RDF API for PHP. In: Proceedings of Workshop on Scripting for the Semantic Web, SFSW 2005, at 2nd European Semantic Web Conference, ESWC 2005, Heraklion, Greece.

[Pan & Heflin, 2004] Pan Z, Heflin J (2004) DLDB: Extending relational databases to support Semantic Web queries. Technical Report LU-CSE-04-006, Department of Computer Science and Engineering, Lehigh University.

Authors' Information

Krassimira Ivanova *– University of National and World Economy, Sofia, Bulgaria*
e-mail: krasy78@mail.bg
Major Fields of Scientific Research: Data Mining

Vitalii Velychko *– Institute of Cybernetics, NASU, Kiev, Ukraine*
e-mail: Velychko@rambler.ru
Major Fields of Scientific Research: Data Mining, Natural Language Processing

Krassimir Markov *– Institute of Mathematics and Informatics at BAS, Sofia, Bulgaria;*
e-mail: markov@foibg.com
Major Fields of Scientific Research: Multi-dimensional information systems, Data Mining

ENHANCED TECHNOLOGY OF EFFICIENT INTERNET RETRIEVAL FOR RELEVANT INFORMATION USING INDUCTIVE PROCESSING OF SEARCH RESULTS

Vyacheslav Zosimov, Volodymyr Stepashko, Oleksandra Bulgakova

Abstract: *The developed technology consists of three main stages: collection of information from a search engine; sifting irrelevant information by the pre-selected features; ranking the obtained results by relevance to a user's request. The ranking model is built with the usage of inductive1 GMDH algorithms. The article describes the effectiveness investigation of the developed technology improving the search relevance of target information on the Internet compared with the Google search engines. When studying, three experiments were conducted with one search request chosen for each experiment. The search results for every request obtained from Google were subsequently processed with the developed technology.*

The first 100 sites from Google SERP were analyzed to compare the relevance level of Google search and that provided with our technology. Outcomes of the experiments are given in the form of circle diagrams showing the percentage of different types of sites in the search results before and after processing it using the proposed technology. The research demonstrates higher effectiveness of the proposed technology compared to Google search: the developed technology allows achieving the search relevance at the level of 80%. Application of this technology will enable more convenient and relevant search of target information on the Internet.

Keywords: *Information search, target information, search engine, search relevance, inductive modeling.*

ACM Classification Keywords: *H.3.5 Information Search and Retrieval - Information Filtering; H.3.5 Online Information Services – Web-based Services*

Introduction

Active artificial promotion of commercial websites to obtain new customers led to the fact that results of a search engine for majority of requests contain large amount of irrelevant

information at the first positions. Search engines algorithms are constantly improving to deal with the artificial promotion of web resources but despite this the search for relevant information remains to be an increasingly difficult task.

For clarity let us distinguish two classes of all the information located on the Internet: the business as well as scientific and technical ones. Below under the commercial information we mean a promotional one provided on a site to attract new customers, visitors, subscribers, etc. to get a commercial gain as a result.

One of possibilities for increasing the efficiency of relevant information search is separation of the whole information on the Internet into commercial as well as scientific and technical according to some predefined attributes. So it is about a solution of two consecutive tasks: sifting irrelevant information in the web and ranking search results using the model built from a specified training sample.

To construct ranking models, a generalized iterative algorithm of the group method of data handling (GIA GMDH) was chosen as an effective inductive modeling tool among a number of methods and algorithms.

Description of the developed technology

For solving the problem of improving the efficiency of relevant information search on the Internet it is necessary to develop a system that provides:

1. High search precision rates, it means the lack of search spam and artificially promoted sites among search results.
2. Search completeness rates not worse than by current search engines.
3. High performance of search results analysis.
4. Wide capacity of software customization by user.

Proposed technology consists of the three main phases: information collecting, sifting of commercial information, ranking the results.

Phase 1. Information collecting. It is not necessary to develop an individual search robot to collect and index data from Web pages because today's search engines cope successfully with the task of data collection and its subsequent indexing. So it is appropriate to use information from the Google search engine database being the most popular and providing opportunity to connect directly to its database using Google API Interface. Google database provides not only a list of sites relevant to the request entered

by a user but also a number of ranking attributes of these sites which will be needed at the ranking phase.

Sites list obtained from the Google database is stored and transmitted for processing to the next phase in which the commercial information is sifting out based on characteristic attributes. Marks received from the Google database with a list of sites are not used on the sifting phase and stored for the ranking phase.

Phase 2. Sifting the commercial information. In the phase of commercial information sifting, a classification model based on the selected set of attributes is represented as a set of decision making rules. For sifting commercial information the DNF-classifier is used in which for a category *C* "commercial information" in course of research there was predefined a number of characteristic attributes $\{a_1^C,\ldots,a_n^C\}$ (where *n* is the total number of attributes) and a set of the site structural elements $\{b_1^C,\ldots,b_m^C\}$ (where *m* is the total number of the elements) containing these attributes.

The classifier is built like such an example:

IF ((a_1^C AND b_1^C) OR

(a_2^C AND b_1^C) OR

…

(a_1^C AND b_m^C) OR

(a_2^C AND b_m^C) OR

…

(a_n^C AND b_m^C))

THEN Commercial information

ELSE NOT Commercial information

The list of the structural elements that are analyzed for the presence of characteristic attributes:

- meta tags, paths to Java-scripts and stylesheet decoration;
- title, meta description, keywords;
- text on the home page;
- navigation elements.

Phase 3. Results ranking. To implement qualitative ranking of sites that remain after the filtering stage, it is necessary to define the weights of ranking attributes obtained from the Google database during data collection phase and get a new ranking model based on these weights.

We do not use the Google rankings data because its algorithms are set up to rank sites taking into account the presence among them a large amount of search spam. Considering this, in Google's algorithm a lot of weights have external attributes (number of external links, domain age, web pages authority, etc.) because it is harder seemingly to forge them than internal attributes. And accordingly minor weights are set to internal attributes (the presence of keywords in the title, number of keywords per page, etc.). Google's algorithm ranks well search results with a lot of search spam but its ranking model should be adjusted. In the developed technology, the search results ranking is implemented according to a new ranking model built using GIA GMDH.

Investigation of the technology effectiveness

The research consists of three experiments. For each experiment was chosen one search request. Selected request was processed at first with the search engine Google and then handled with the developed technology. To simplify of the experiments description, only first 100 sites from Google SERP were analyzed. In what follows we compare the performance of Google search and the developed technology.

Experiment 1. Investigation of the technology effectiveness for the "Information Security" request for search engine google.com.ua.

The tested search query: "Information Security".

Fig.1 shows the percentage of different types of sites among the top 100 results found for this request.

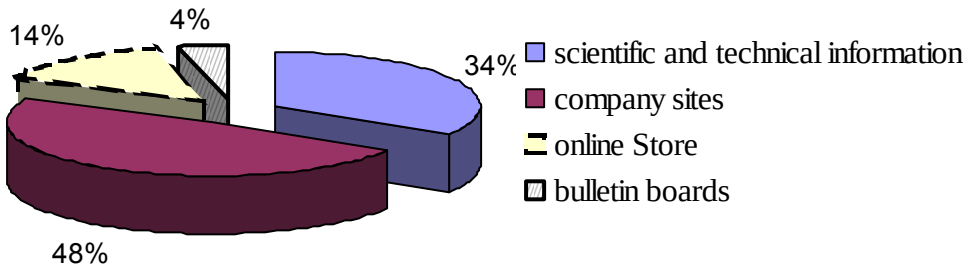


Fig. 1 Percentages of different types of sites from google.com.ua SERP for the "Information Security" request

Fig. 1 demonstrates that the target scientific information forms only 34% of the total number of sites found. The remaining 66% were mostly sites of companies that provide services for the information protection and some online stores selling products to protect information.

Then we processed the same query using our technology. Fig.2 illustrates the obtained results.

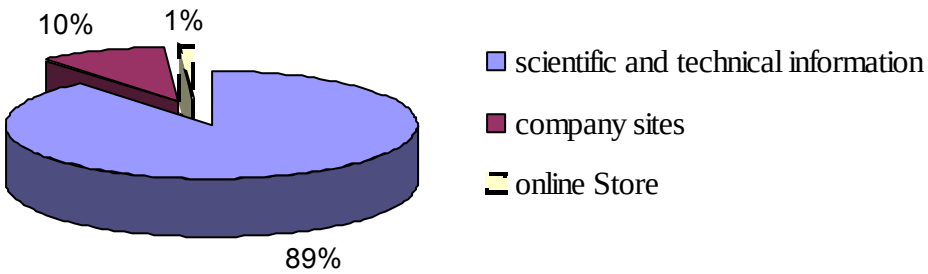


Fig. 2 The percentage of different types of sites from google.com.ua SERP for the "Information Security" request after additional processing

According to Fig. 2, the percentage of scientific information have raised to 85%. This is almost three times better than Google results and updated search results include only 15% of commercial sites. At the same time any site containing the scientific information was not wrongly ignored. This means that the performance of the search delivery was not reduced.

Experiment 2. Investigation of the technology effectiveness for the "Programming 1C" request for search engine google.com.ua

The tested search query: “Programming 1C“.

Fig.3 shows the percentage of different types of sites among the top 100 results found for this request.

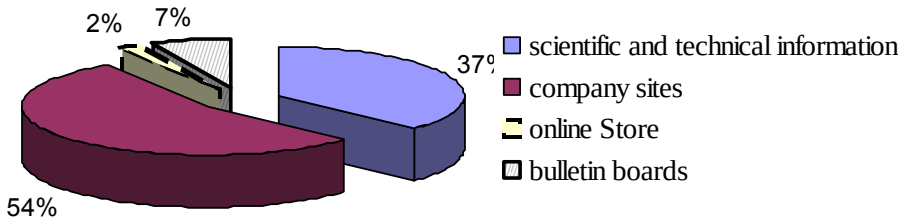


Fig. 3 The percentage of different types of sites from google.com.ua SERP for the ”Programming 1C” request

Fig. 3 shows that the requested scientific information keeps only 37% of the total number of sites found. The remaining 63% were mostly sites of companies that provide services for the 1C programming, bulletin board and some online stores selling products of 1C Company.

Fig.4 illustrates the percentage of various sites categories in the remaining results after processing the same query using our technology.

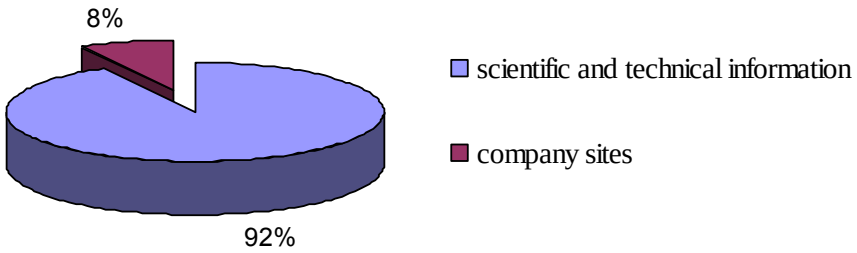


Fig.4 The percentage of different types of sites from google.com.ua SERP for the ”Programming 1C” request after processing

Figure 4 shows that the share of scientific information grew to 92%. This is almost three times as much as Google results, includes only 8% of commercial sites and any site containing scientific information was not wrongly aborted. Hence the performance of the search output was not reduced.

*Experiment 3. Investigation of the technology effectiveness for the request "BOBCAT engine structure" (*bobcat is an American manufacturer of forklifts*) for search engine google.com.ua*

The tested search query: “BOBCAT engine structure “.

Fig.5 shows the percentage of different types of sites among the top 100 results found for this request.

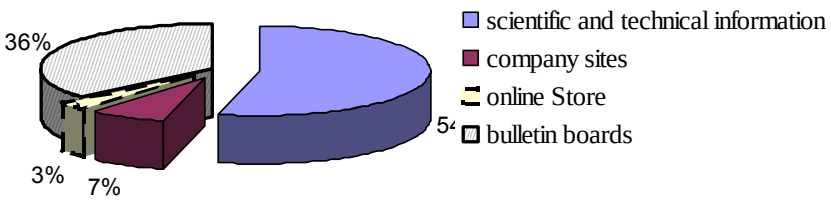


Fig. 5 The percentage of different types of sites from google.com.ua SERP for the ” BOBCAT engine structure” request

As it is evident from Figure 5, the searched scientific information makes up only 54% of the total number of sites found. The remaining 46% were mostly bulletin board, sites of companies that provide services for BOBCAT engines, and some online stores selling engines.

Than the same query was processed using our technology and Fig.6 illustrates the share of various sites categories among the obtained results: the percentage of the scientific information increased to 83% and only the rest 17% of commercial sites. This is more than half much again as Google results and any site containing scientific information was not wrongly excluded, so the performance of search message was not reduced.

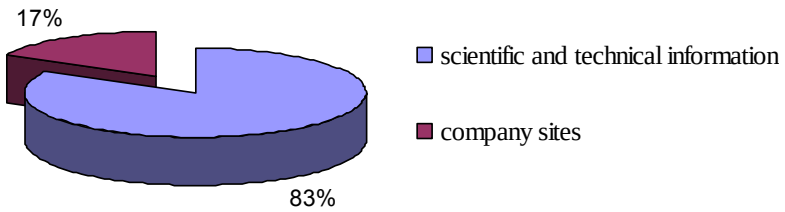


Fig. 6 The percentage of different types of sites from www.google.com.ua SERP for the ”BOBCAT engine structure” request after processing

Table 1 shows the results of application of the developed software system for solving the problem of increasing the efficiency of scientific and technical information search in two state organizations: Ukrainian Radio Engineering Institute (UREI), and Mykolaiv National University (MNU).

Ranking performance of search results was estimated as an average number of links found for a user's request. This evaluation method of ranking is appropriate though not the only possible.

Table 1. Application results of the developed software system

Organization	Search accuracy performance in %		Percentage of not sifted commercial sites	Percentage of wrongly sifted relevant sites
	Search engine	Developed software system		
UREI	83%	97%	3%	0,4%
MNU	54%	85%	8%	0,5%

Table 2 shows the results of comparing the ranking performance by the Google search engine model and that built by GIA GMDH model.

Table 2. Ranking performance results

Average number of visited links		
Without sifting the irrelevant information	After sifting the irrelevant information	
Google ranking model	Google ranking model	Using GIA GMDH ranking model
12	10	5
17	11	6

Building of Google search engine ranking model

To check the quality of bult using GIA GMDH ranking models we decided to rebuilt Google ranking model. For this we simulated (have found the model), Google ranking process of web resources for the search request "web programming".

Input variables:

For the experiment, we selected the first 50 sites from Google search engine result page (SERP).

Model quality was evaluated on a sample *B* as the value of the regularity criterion AR.

$$AR = \left\| y_B - X_B \hat{\theta}_A \right\|^2 \quad (1)$$

The matrix X contains 42 variables-factors that numerically characterize each site and divided into two parts: 2/3 - study A, which is used for coefficients estimation, the other 1/3 - test sample B. Matrix columns are corresponding to the values of the factors, and lines - are corresponding to the Web resource.

To simulate the ranking process of web resources search results, the following attributes were used:

x_1 – keywords number on site;
x_2 – keywords number on page;
x_3 – the ratio of total words number on site to the keywords number on site;
x_4 – the ratio of total words number on page to the keywords number on page;
x_5 – Google Page Rank;
x_6 – topic's popularity;
x_7 – number of requests for a particular keyword in a given period of time;
x_8 – total number of web pages;
x_9 – amount of text on site;
x_{10} – amount of site;
x_{11} – amount of web page text;
x_{12} – age of site;
x_{13} – the keyword presence in URL of the site (domain name);
x_{14} – frequency of updating site information;
x_{15} – the last update;
x_{16} – number of images on site;
x_{17} – number of multimedia files;
x_{18} – the presence of alt-tags for images;
x_{19} – alt-tags length (in symbols);
x_{20} – usage of frames;
x_{21} – site language (Russian or foreign);
x_{22} – keywords font size;
x_{23} – keywords font weight;

x_{24} – the distance between keywords (in symbols);
x_{25} – written in capital letters or not the keywords;
x_{26} – How far from the beginning of the web page are keywords;
x_{27} – the presence of keywords in title;
x_{28} – the presence of keywords in meta-tags;
x_{29} – the presence of file «robot.txt»;
x_{30} – site's location;
x_{31} – comments in source code;
x_{32} – what type of pages, each page relates: html or asp;
x_{33} – the presence of flash files;
x_{34} – the presence of the same pages on site;
x_{35} – matching of site keywords to the search engine directory partition in which the site is;
x_{36} – the presence of "noise words" ("stop words");
x_{37} – total number of links;
x_{38} – number of internal links;
x_{39} – number of external links;
x_{40} – site depth;
x_{41}– number of external links with keywords in title;
x_{42} – Yandex citation index.

Output variable: ***y*** – web resource position among the ranking results.

Model accuracy was calculated by the determination coefficient formula:

$$R^2 = \frac{\sum_{i=1}^{n}\left(\hat{y}_i - \bar{y}\right)^2}{\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2} 100\%, \qquad (2)$$

where $\bar{y}$ is an average value, $\hat{y}_i$ is the model output.

Using the generalized GMDH algorithm, the following model was constructed that describes the search engine ranking process of web resources:

$$y = 3{,}24 + 2{,}71x_3 + 0{,}12x_4 + 0{,}00003x_7 - 2{,}69x_{12} + 0{,}012x_{22} - 14{,}8x_{27} - \\ - x_{28} - 27{,}29x_{35} + 4x_{40} - 0{,}006x_{41} - 7{,}89x_5x_6 + 0{,}06x_{14}x_{15}^2 + 0{,}002x_{37}x_{38}x_{39} \qquad (3)$$

$AR(A) = 2{,}48$; $AR(B) = 3{,}51$, $R^2 = 92\%$

Table 3 shows the results of sites ranking using model (3).

Table 3. Results of sites ranking using model built with GIA GMDH

Place in google.com.ua	Values by GMDH	Rounded results
1	1,23	1
2	1,89	2
3	4,01	4
4	4,21	4
5	4,89	5
6	6,02	6
7	6,78	7
8	8,00	8
9	8,52	9
10	9,33	9
…	…	…
21	21,23	21
22	22,49	23
23	22,85	23
…	…	…
32	33,56	34
33	33,56	34
34	33,68	34
…	…	…
57	57,22	57
58	58,15	58
…	…	…
99	98,95	99
100	99,56	100

Analysis of the built model shows that main influence on the google ranking model has the following 16 factors:

x_3 – the ratio of total words number on site to the keywords number on site;
x_4 – the ratio of total words number on page to the keywords number on page;
x_5 – Google Page Rank;
x_6 – topic's popularity;
x_7 – number of requests for a particular keyword in a given period of time;

x_{12} – age of site;
x_{14} – frequency of updating site information;
x_{15} – the last update;
x_{22} – keywords font size;
x_{27} – the presence of keywords in title;
x_{28} – the presence of keywords in meta-tags;
x_{35} – matching of site keywords to the search engine directory partition in which the site is present;
x_{37} – total number of links;
x_{38} – number of internal links;
x_{39} – number of external links;
x_{40} – site depth;
x_{41} – number of external links with keywords in title;

After analyzing these factors one can say that main influence on Google ranking has the external factors (x_5, x_6, x_7, x_{12}, x_{35}, x_{39}, h_{41}) as compared to internal ones.

We verify correctness of constructed model (3) for other search queries:

- «omelet recipe»
- «buy notebook Kiev»
- «expert systems»

Table 4 shows the results of comparing web resources ranking by Google and built using GIA GMDH models

Table 4. Web resources ranking results

Place in google.com.ua	Values by GMDH		
	«omelet recipe» / rounded result	«buy notebook Kiev» / rounded result	«expert systems» / rounded result
1	0,83 / 1	1,02 / 1	0,78 / 1
2	1,91 / 2	2,11 / 2	2,02 / 2
3	3,09 / 3	3,56 / 4	3,01 / 3
…	…	…	…
15	14,89 / 15	15,08 / 15	14,98 / 15
16	16,02 / 16	16,06 / 16	14,99 / 15
17	16,78 / 17	17,21 / 17	14,99 / 15
…	…	…	…

Table 4 continued. Web resources ranking results

Place in google.com.ua	Values by GMDH		
	«omelet recipe» / rounded result	«buy notebook Kiev» / rounded result	«expert systems» / rounded result
21	19,52 / 20	21,11 / 21	21,03 / 21
22	21,33 / 21	22,13 / 22	21,89 / 22
23	23,01 / 23	24,05 / 24	22,99 / 23
…	…	…	…
37	37,91 / 38	36,99 / 37	36,89 / 37
38	37,95 / 38	38,00 / 38	38,01 / 38
39	38,23 / 38	38,78 / 39	39,05 / 39
…	…	…	…
62	63,06 / 63	62,12 / 62	62,13 / 62
63	63,56 / 64	63,42 / 64	62,58 / 63
64	64,18 / 64	64,01 / 64	64,02 / 64
…	…	…	…
77	77,02 / 77	76,01 / 76	78,00 / 78
78	78,11 / 78	77,72 / 78	78,32 / 78
…	…	…	…
100	99,86 / 100	100,56 / 101	100,01 / 100
R^2	87%	95%	93%

Table 4 shows that the model built using GIA GMDH accurately follows the Google ranking of web resources and can be used to further investigation of ranking methods.

Conclusion

Results of the experiments described in this article shows that the developed technology allows achieving high precision of search results containing more than 80% of relevant scientific information. This is due to using the results of Google search engine and applying inductive GMDH algorithm. All this indicates that the usage of this technology will

allow making search for scientific and technical information on the Internet more convenient, simple and precise.

Application of the developed software system for solving applied problems improves greatly the accuracy of relevant information search necessary for the research.

The usage of the inductive algorithm for building ranking models is efficient. Models built using GIA GMDH helps greatly reduce the time needed for search of relevant information.

High accuracy of the constructed Google ranking model proves the effectiveness of a generalized iterative algorithm GMDH for solving such kind of problems.

Bibliography

[Stepashko] Stepashko V., Bulgakova O., Zosimov V. Performance of Hybrid Multilayered GMDH Algorithm. – Proceedings of the III International Workshop on Inductive Modelling IWIM-2011, 5-9 July 2011, Kyiv-Zhukyn, Ukraine. – Kyiv: IRTC ITS NANU, pp 109-113, 2011.

Authors' Information

Viacheslav Zosimov *– Lecturer of Mykolaiv V.O. Suhomlynsky National UniversityI, Ukraine; e-mail: zosimovvv@bk.ru*

Major Fields of Scientific Research: Web-technologies, Information Search and Retrieval.

Volodymyr Stepashko *– Head of Department for Information Technologies of Inductive Modeling of IRTC ITS, Professor, Dr Sci, P.A.: 40, Akademik Glushkov Prospect, Kyiv, Ukraine, 03680; e-mail: stepashko@irtc.org.ua*

Main Fields of Scientific Research: Data Analysis Methods and Systems, Knowledge Discovery, Information Technologies of Inductive Modelling, Group Method of Data Handling (GMDH)

Oleksandra Bulgakova *– PhD, Associate Professor of Mykolaiv V.O. Suhomlynsky National University, Ukraine; e-mail: sashabulgakova@list.ru*

Major Fields of Scientific Research: Information Technologies of Inductive Modeling.

INTELLIGENT AGENTS AND MULTI-AGENT SYSTEMS

AN AGENT–ORIENTED ELECTRONIC MARKETPLACE FOR MODELING AND SIMULATION OF DYNAMIC PRICING MODELS BUSINESS LOGIC

Jacek Jakieła, Paweł Litwin, Marcin Olech

***Abstract**: The main goal of the research that preliminary results have been presented in this paper is to develop an agent–oriented electronic marketplace for modeling and simulation of dynamic pricing models, i.e. models in which the price of the item is allowed to fluctuate as supply and demand in a market change. The work provides an overview of forms of dynamic pricing models, with particular emphasis on auctions. After that, the main rationale behind using Multi-agent Systems approach for modeling and simulation of complex business structures has been shown. Then the development process of an electronic marketplace, including agents' architecture as well as implementation environment selection, structure and business logic of e-marketplace have been presented. Last part of the paper comprises conclusions and further research plans.*

***Keywords**: Agent-Based Models, Simulation for MAS development, Agent-oriented Marketplace Design, Multi Agent Based Modeling and Simulation, Dynamic Pricing Models*

***ACM Classification Keywords**: I. Computing Methodologies; I.2 Artificial Intelligence; I.2.11 Distributed Artificial Intelligence; Multi-Agent Systems*

Introduction

The opportunities of using *Multi-Agent Systems* (MAS) and *simulation* are numerous. They have already been applied and coupled in many application domains and for different purposes [Weyns et al., 2009].

The paper describes the first steps undertaken in the new area that is being investigated by authors, but which is the continuation of work that results have already been published

in [Jakieła et al., 2009][Jakieła, Litwin, Olech, 2010a][Jakieła, Litwin, Olech, 2010b][Jakieła, Litwin, Olech, 2011][Jakieła, Litwin, Olech, 2012]. The research conducted so far was focused on application of *Multi-Agent Based Simulation* (MABS) to Supply Chain Management as well as the analysis of Supply Chain behavior (e.g. bullwhip effect analysis).

The goal of the newly started research is to develop an agent-oriented electronic marketplace that will be used for analysis of dynamic pricing models, the models in which price of the item is not fixed but may be changed before transaction is finalized. Activities related to price determination are part of many business processes conducted by firms which are a part of supply chains. For example at the buy-side firm is executing transactions with suppliers, while at the sell-side it interacts with customers. All these activities are increasingly subject to automation as a part of electronic marketplaces.

Automating markets leads to many benefits. One is cost saving from automating functions of non-computational markets such as searching for goods and potential trading partners or price discovery automation. Another benefit would be the ability to extend markets in time and geographic scope by conducting them over networks [MacKie-Mason et al., 2006]. The greatest potential however may be related to the opportunity to deploy market mechanisms that cannot effectively operate without computer automation e.g. dynamic pricing.

When automating market related activities one can take into consideration different alternatives for business logic that will finally be implemented. For example, when the price discovery mechanisms are supposed to be implemented in the form of dynamic pricing model, then *which model is the best choice?* The solution to this problem would be to prepare the experiments which results will be used in the process of business logic selection. Therefore the goal set for the research is to use multi-agent based simulation as a test-bed for this kind of analysis.

The paper is structured as follows. Firstly the advantages of dynamic pricing models applications have been described. After that, the rationale behind using agent paradigm has been explained. Then the problem of e-marketplace design has been characterized. Finally the development process of agent oriented electronic marketplace prototype has been presented in detail.

Virtual Dynamic Pricing as an Efficient Way of Price Discovery

Revenue generation is the core ingredient of every business model used by the contemporary firms. It consists of two main elements: *market mechanisms for price determination* and *revenue sources*. In every firm operating as a part of the supply chain

there are many processes where transaction is executed. It may be the buy-side of the organization where procurement and inbound logistics processes take place. On the sell-side every firm is dealing with intermediaries and customers selling and distributing to them products and services. Wherever transaction is executed, the price for the transaction has to be accordingly determined. There are three commonly used mechanisms of price determination: *menu pricing*, *dynamic pricing* and *bartering*.

In the classical Old Economy context, the basic mechanism is *menu pricing* also known as *fixed pricing*. It works according to the logic, where seller sets the price and the buyer may take it or leave it. This model is used by nearly every retail store where prices for products are fixed and cannot be changed. Although menu pricing has been working for many years it has two main shortcomings. First one is related to the situation in which buyer may be willing to pay more than the price set by the seller. It could be said that seller is leaving money on the table. The second issue is that the menu price is too high and cuts off many buyers. They would have bought the product at a lower price. The stickiness of the prices is the result of two main factors. Firstly, it is not easy to detect changes in consumer preferences quickly enough to effect price changes. Secondly, in an off-line context it is quite hard to implement price changes [Afuah et al., 2002].

These problems may be solved by dynamic pricing which business logic is encapsulated in the functionality of e-marketplace used on-line by buyers and sellers. The first issue is that the price may be changed according to specific protocol. What is more, with the Internet the customer preferences can be detected more easily and cost of changing prices is lower because they all are virtual.

In dynamic pricing model the price is not fixed but may change over time what overcomes the disadvantage mentioned before which lets some customers get away with the price that is less than they would be willing to pay and misses out on customers who would prefer to pay less.

Dynamic pricing may appear in several forms. There are many classifications. In the most popular one, the form of dynamic pricing depends on number of buyers and sellers involved in the process. When there are one buyer and one seller, the pricing is based on negotiation, bargaining or bartering which are the oldest forms that have been practiced for many generations in markets, usually open-air. The final price will be determined by bargaining power of each party, business factors and supply/demand in the item's market [Turban et al., 2011].

In case there is one seller and many potential buyers or one buyer and many potential sellers the pricing model is called an *auction*. If there is one seller and many buyers the auction takes a form of *forward auction*, in which seller entertains bids from multiple

buyers. In the *reverse auction* there is usually one buyer and many sellers. Buyer places an item she wants to buy for bid and potential supplier bid on the item, reducing the price sequentially. Reverse auctions are primarily a Business-to-Business (B2B) pricing mechanism.

The auction is driven by *auction protocol* which determines when transaction is finalized, who will get an item and how much the winner will pay for it.

In *English auction* seller begins by calling out a starting (most often low) price which is gradually raised, apparently to all buyers, usually in small increments. The auction stops after predefined period of time ("deadline" auctions) or when there is only a single bidder who is still interested. Finally the item is sold to the highest bidder.

The *Dutch auction* is open descending price counterpart of English auction. Seller begins with initial price and then gradually decreases it. Starting price is high enough so nobody is interested in buying the item at that price. The price is lowered until some bidder indicates her interest and the item is sold to her at the given price.

In the *sealed-bid first-price auction* bidders submit bids in sealed envelopes. The highest bid wins and the winner gets the item and pays what she bid. The *sealed-bid second price auction* is the process in which bidders submit bids in sealed envelopes and the highest bidder wins but this time she pays second–bid [Milgrom, 2004].

According to the research goals, all of the mentioned auction protocols are supposed to be implemented as business logic of e-marketplace that will finally be tested with the use of agent–oriented simulation. Next section describes the main rationale behind using agent paradigm for this purpose.

Agent-oriented Simulation as a Test-Bed for Dynamic Pricing Business Logic of e-Marketplace

The MABS approach to modeling complex business structures has been becoming more and more popular; its models as well as their advantages have been widely described in [North et al., 2007][Weyns et al., 2009]. Contributions to the MABS domain are periodically published among others in Springer's LNCS and LNAI series [Jakieła, Litwin, Olech, 2010b][Jakieła, Litwin, Olech, 2012].

Multiagent-based simulation (MABS) can be defined as the modeling and simulating real world system or phenomena where the model consists of agents cooperating with one another in carrying out tasks and achieving collective goals. The advent of multi-agent based modeling has introduced an important innovation: behavior of complex systems with many active entities can be simulated by modeling individual entities and their interactions. Importantly, the operation of the system doesn't need to be defined a priori

as set of equations, terms or logical statements, but the whole behavior emerges from individual objects behaviors, their interactions and impact of the environment.

As this paper focuses on agent paradigm application to price discovery on e-marketplace, it is important to look at what have been done so far in this domain. Of course it is possible to design markets without agents. As Marks suggests, in such case one has to have market with demand and supply schedules and what is more, economic efficiency is maximized at the output level where marginal value equals the marginal unit cost, no matter how the social surplus is divided between buyers and sellers [Marks, 2006]. There is only one drawback, as this is optimization the problem has to be well defined. Because of several possible design trade-offs and emergence of unforeseen performance in the system, modeling market system as evolving system of autonomous, interacting agents is increasingly employed. This thesis may be backed by opinion provided by LeBaron who states that agent-based models are well suited to examining market design because they can produce large amount of data and allow testing of market design in a heterogeneous, adaptive environment [LeBaron, 2006]. As many applications have already proved, using agents in market design is quite reasonable choice. The agent paradigm have already been used in analysis of market's micro–structure [Audet et al., 2002], examination of tick sizes [Bottazzi et al., 2003], analysis of double auction process where buyers and sellers, equipped with heuristic rules (belief-based learning) are trying to assess the probabilities that they offers to buy/sell will be accepted, given market history [Gjerstad, 2004]. Agent applications may also incorporate genetic algorithms that are used to encode market decisions that agents can make and to find the optimal one or to encode beliefs about changes of prices from period to period in order to find optimal consumption/savings allocations and market–clearing prices [Arifovic, 1994][Duffy, 2006] [Bullard et al., 1999].

All mentioned pros backed the main goal of the research work which is *to use agent-oriented electronic marketplace as a test–bed for analysis of dynamic pricing models in order to determine which business logic should be encapsulated in the multi-agent system supporting price discovery in procurement and in-bound logistics processes in supply chain*.

The sections below present the development process with all design decisions that had to be made such as: selecting agents' architecture and implementation environment, defining market structure, and finally designing individual agents. In the description that follows only English auction model has been taken into account, but the development process for other models follows the same structure and logic.

The Development Process of Electronic Marketplace in Detail

Multi-Agent Systems are today considered as an interesting way of understanding, modeling, designing and implementing different kind of distributed systems. From the perspective of planned research, the application of MAS has two aspects. Firstly, as Parunak claimed, MAS represent modeling alternative, compared to equation based modeling, for representing and simulating real world or virtual systems which may be decomposed in interacting individuals [Parunak et al., 1998]. At the same time MAS could be used as a programming paradigm to develop software systems. Agent paradigm is especially suited for solutions in which global control is hard or not possible to achieve [Zambonelli et al., 2002]. The modeling aspect requires good understanding of architecture of modeling constructs – agents as well as the whole system – MAS, and the implementation needs–proper environment in which it will be conducted. These two issues will be discussed in the following sections.

Selecting an Agents' Architecture

There are two main approaches for analyzing agent's architecture: *reactive* and *cognitive*. In reactive approach only the perception–action also known as stimuli–response component is considered. The cognitivist approach relies on the mental issues and assumes that an agent has mental states as well as the partial representation of the environment and other agents. There is also third approach called hybrid, trying to get together first two.

The best-known cognitive architecture, which has been employed in the process of e-marketplace design and development, is the BDI (*Belief-Desire-Intention*) architecture. The main quality of BDI architectures is to create a behavior which mimics that of a rational human being [Rao et al., 1992].

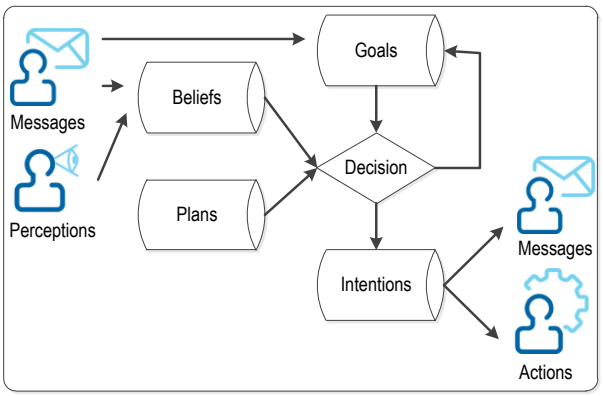


Fig. 1. The elements of BDI architecture

In this architecture there is an assumption that agents infer and act on the basis of knowledge that is represented with a symbolic formalism. What is more, agent possesses mental states such as *beliefs*, *goals* (desires), and *intentions*. Agent acts intentionally in order to achieve goals and this behavior is based on beliefs about world states. Every agent is equipped with library of plans. The plan works as a recipe for a goal achievement and contains a set of actions. When the agent commits itself to achieve the specific goal with the use of specific plan then the intention is set. All the mental states are dynamic what means that are updated during agent's life cycle according to environment's changes and messages from society. The figure 1 illustrates all of the architecture basic building blocks and relationships among them [Ferber et al. ,2009].

As the figure 1 shows beliefs are formed based on perceptions concerning environment changes and messages received from other agents. Each agent may initially have one or more goals. In order to achieve goal, decision component is trying to select plans that are compatible with agent beliefs and executes actions included in plans. Plan that has been chosen for execution is transformed into intentions that finally result in environmental as well as communication actions. The whole process is controlled by BDI engine which works according to logic embedded in implementation environment.

Selecting the Implementation Environment

In recent years, there is extremely rapid increase in amount of research being done on agent-oriented development platforms and programming languages. As a consequence of agent's architecture selection, BDI based implementation environment has been chosen. Presented environment consists of language which is a variant of AgentSpeak programming language and its interpreter called *Jason*. In the following paragraphs, the basic language constructs have been described as well as the structure of an interpreter.

AgentSpeak has its roots in logic programming. Therefore basic representation units, which are used to denote mental attitudes of agents, are predicates. The main language constructs, as in case of BDI architecture, are related to mental states and denote beliefs, goals and plans. *Beliefs* are used to represent information agent stores about environment, other agents and itself. Interesting fact about beliefs in Jason is that they are annotated and therefore may be maintained on the meta-level. There are three main annotations such as: *percept*, *self* and *agent name*. Percept is used to denote information from the agent sensors (received from environment). Self means that the belief is created by agent as a *mental note*. Agent name suggests that source of the belief is other agent.

Goals represent the state of affairs the agent strives for. The representation of goal is the same of a belief expect that it is prefixed by symbol "!". *Plans* constitute courses of

action an agent will execute in order to achieve its goals or to react to changes in its environment. Every agent has the library of plans that determines its behavior.

The plan is structured as presented below.

```
triggering_event : context <- body.
```

The *triggering_event* represents event that will be handled by plan. *Context* describes circumstances under which the plan is suitable to handle the event. The *body* is a sequence of actions that will be executed or new goals for the agent to achieve. The agent behavior may change over time if new plans are acquired during the communication with other agents.

Agent acts according to agent's program which specifies initial beliefs, initial goals of the agent, and the plans in the plans library available to agent when it starts running.

Agent program is executed by Jason interpreter which uses a number of important data structures that allow to implement BDI agents. Figure 2. shows the main components of the Jason interpreter [Bordini et al., 2009].

Belief base is responsible for gathering and storing all the beliefs agent possesses and is updated in every reasoning cycle on the basis of all percepts received by from environment as well as other agents.

Events result from changes in beliefs and goals. Beliefs may be updated and new goals set or received from other agents as a part of the delegation process. Events trigger execution of plans, provided that event matches the *triggering_event* and is applicable the time is chosen.

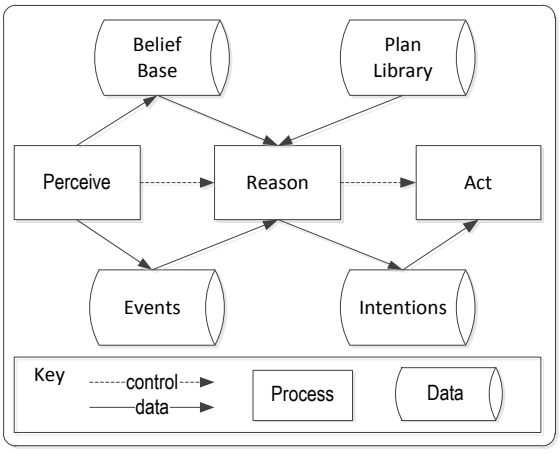


Fig. 2. Simplified view of Jason interpreter components [Weyns et al., 2009]

Plans library stores agent's know-how and includes all the plans written for an agent in AgentSpeak code. In case of simple agents the plan library remains unchanged; however it is possible to change agent behavior by plan exchange using speech-act based communication. Plans may be labeled and annotated what enables meta-level advanced processing in a selection function.

Set of intentions that are created every time the change in environment is perceived by an agent and there is applicable plan for an event. Each intention is a stack of partially instantiated plans and represents a "focus of attention" for the various tasks currently being carried out by an agent. Intention can be dropped or revised.

The body of the plan is composed of actions. Unlike actions, *internal actions* don't change the state of the environment. They are mechanism to allow legacy code to be referenced from the high–level agent reasoning as defined by AgentSpeak code. There are several predefined internal actions provided by Jason to help with various programming tasks. All internal actions start with the "." character [Weyns et al., 2009].

The structures presented above are all essential for BDI agents. Of course there are various other structures used by Jason interpreter but due to the space restrictions of the paper it is not possible to go into enough detail. Complete account of the Jason implementation is available in [Bordini et al., 2007]

The Design of Electronic Marketplaces

Markets play central role in the economy, facilitating the exchange of information, goods, services and payments. Three main functions of every market are: matching buyers and sellers, facilitating the exchange of information, goods, services and payments associated with market transactions, and providing institutional infrastructure (legal and regulatory framework) which enables the efficient functioning of the market [Bakos, 1991].

The emergence of electronic marketplaces introduced several changes in processes used in supply chains, which resulted in [Turban et al., 2011]:

- greater information richness of the transactional environment,
- lower information search costs for buyers,
- diminished information asymmetry between sellers and buyers,
- the ability of buyers and sellers to be in different locations,
- better, more efficient mechanisms for price discovery.

From the perspective of this paper especially important is the last issue from the list above. The goal is to design an electronic marketplace used for analysis of dynamic pricing models.

Designing markets is rather a new discipline. Marks lists the following five examples of designed market [Marks, 2006]:

- *Simulated stock markets* for new financial instruments and derivatives. They have been developed after Black and Scholes solved the problem of pricing options.
- *Markets for pollution emissions* where market mechanisms are used to control emission of sulfur dioxide and carbon dioxide.
- *Markets for electro–magnetic spectrum* where bands of local spectrum to be used for new communication technologies are sold.
- *Markets for electricity exchange* where several types of new market mechanisms have been introduced because classical ones were not appropriate.
- *On-line markets* also known as *e-marketplaces* that provide opportunities to buy and sell on-line with the use of Internet infrastructure. There are several types of them. Major Business-to-Consumer (B2C) e-marketplaces are *virtual storefronts* and *e-malls*. Business-to-Business (B2B) e-marketplaces include *sell-side* and *buy-side* e-marketplaces and *exchanges*.

The business logic that is supposed to be tested on implemented agent–oriented e-marketplace may be used in the design of sell-side, buy-side e-marketplaces as well as B2B exchanges.

The Structure and Business Logic of e-Marketplace

According to MacKie-Mason there are mainly three stages, parties must go through in order to execute a transaction [MacKie-Mason et al., 2006]: (1) *connecting* that is responsible for discovery of opportunity to engage in a market interaction; (2) *interaction* which is the negotiation of terms; (3) *exchanging* – execution of terms of the finalized transaction. The paper focuses mainly on the first two stages. Connecting is realized with the use of communication protocols the agents use. Negotiation of terms, as will be shown later, is done according to selected dynamic pricing models protocols.

In order to organize presentation of the development process, the framework of marketplace system has been used, that consists of the following elements:

- *Marketplace system* – the society of agents that participate in resource allocation problem, together with market mechanisms used during interaction in order to finalize a transaction.
- *Mechanisms* – the rules specifying permissible actions and the outcomes as a function of agent actions.
- *Market participants* – agents with BDI architecture that represent autonomous decision making locus in the system of many decision making entities.

Depending on the dynamic pricing model analyzed, agents operate according to some decision rules (market mechanisms).

The general structure of the marketplace system is quite simple. It consists of buyer and seller sides. Seller–side is composed of seller agents and buyer–side includes buyer agents. The system has been parameterized what enables to easily change the number of agents.

Because the main aim of the marketplace is to analyze the business logic related to dynamic pricing models, these mechanisms are based on auctions protocols. Protocols selected are those, the most often used nowadays. There are English Auction, Dutch Auction, First-Price Sealed–Bid Auction (FPSB) and Second–Price Sealed–Bid Auction (SPSB). All of them have shortly been described in the section entitled *Virtual Dynamic Pricing as an Efficient Way of Price Discovery*.

Every protocol is characterized by the set of parameters that have been presented in the table 1. During simulation the values of parameters are supposed to be randomly generated. Specific distribution may be selected by modeler. In the default settings *uniform distribution* is used.

Table 1. Parameters of auction protocols used

Protocol	Reservation price	Initial product price	Reaction time	Duration of the auction	Decreasing price's time	Decreasing price's value
English	*Yes*	*Yes*	*Yes*	*Yes*	*No*	*No*
Dutch	*Yes*	*Yes*	*Yes*	*No*	*Yes*	*Yes*
FPSB	*Yes*	*No*	*Yes*	*Yes*	*No*	*No*
SPSB	*Yes*	*No*	*Yes*	*Yes*	*No*	*No*

The sell–side of the marketplace works according to the following rules:

- Seller agent is supposed to sell products on the marketplace.
- Seller agent has *reservation price* established, that is a minimal acceptable value of transaction, and the *reaction time* that determines when the auction winner will be announced and when the product will be offered on the marketplace.
- Seller agent also knows the *duration of the auction* in case of Sealed-bid auctions as well as English auctions – after specified period of time goes by, the system closes an auction.
- Seller agent sets the *initial product price* for English and Dutch auctions. This is the value the potential buyers start to bid with.

- Seller agent sets two additional parameters for Dutch auction which are related to the velocity of price decrease and the value which is used to decrease the price.

The buy–side of the marketplace operates with regard to the following assumptions:

- Buyer agent is supposed to buy a product offered on the marketplace.
- Buyer agent has *private valuation of product* established which is the maximum price she is willing to pay, and the *reaction time*.
- Buyer agent has two additional parameters for English auction. The first one concerns the value which is used to increase the product's price. Second one regards the time after which the buyer will re-bid.
- Buyer agent bidding process consists of the following steps:
    - Checking all open auctions.
    - Evaluating all open auctions utility function values.
    - Removing all auctions that do not meet established criteria (the utility function value is lower than the threshold that has been set).
    - Selection of the auction with highest utility function value.
    - Bidding until the auction is open.

The business logic presented above has been implemented in the e-marketplace. As the seller side is quite simple, next sections describe mainly implementation details of buyer agents.

Buyer Agents Implementation

Buy–side of the e–marketplace consists of two types of agents. The first type is *dummy agent* that operates randomly, according to parameters' distributions selected by modeler. The second category is *smart agent* which is able to use different tactics during the bidding process. All agents' behavior is driven by BDI interpreter according to agents plans implemented. Simple business logic, as in case of dummy agents has been developed with the use of *AgentSpeak*. More complex behaviors that required sophisticated calculations have been implemented in Java language. Thank to flexible architecture of *Jason* interpreter as well as Java interfaces it has, these two implementations can be relatively easily put together and fully integrated.

Dummy Agent Implementation

The operation of dummy agent consists of several activities carried out as internal and external actions. The first action is responsible for setting the parameters presented in the table 2.

Table 2. Dummy agent parameters

Parameter's name	Value Range	Distribution
Reaction time	(5; 8)	Uniform
Product's max price	(100;250)	Uniform
Ace value	⟨1;20)	Uniform
Ace interval	⟨1;5)	Uniform

After parameters' values are generated the beliefs are created and saved in the agent's beliefs base.

In the next step external action causes that list of all auctions recorded in the system is saved and made available to the agent. Auctions have been divided into two groups: open and closed. In addition all open auctions are counted and the result of this operation is stored in agent's beliefs base. The figure 3 presents example beliefs base of an agent after the operation is executed.

engAuction(0.6578,1,sellerEng2)

engAuction(closed,5,sellerEng4)
engAuction(closed,2,sellerEng3)
engAuction(0.2413,3,sellerEng1)
engAuction(0.6772,4,sellerEng5)

Fig. 3. The agent's beliefs base content

The first *engAuction* predicate argument is the value of utility function for auctions that are open or the information that auction has already been closed. The second is an *auction ID* and the third is ID of a seller.

Auctions with utility function value smaller than predefined threshold should be removed. Therefore internal action has been defined for deletion of such auctions from the agent's belief base. When all such auctions have been removed, the number of open auctions agent can bid for has been calculated. Next, the check is done if there are any auctions we can choose from. Finally the internal action finds the maximum element in the list which represents the best option.

The last activity the agent undertakes is to bid for selected auction. Bidding action has been implemented in the environment which is responsible for managing offers. The action ends after the agent receives *noAuction* belief from the system.

The bidding process is conducted according to the following steps:

1. Is the auction open? If it is, then go to step 2; if it is not, go to step 7.

2. Does the last offer is my offer? If it does, go to step 8; if it does not, go to step 3.
3. Calculate value of the bid.
4. Does the calculated value is greater than my private product's estimation value? If it does, go to step 9; if it does not, go to step 5.
5. Place an offer, store transaction information in the system, print information to the log file.
6. Return to the internal action (run the bidding process once again).
7. Print message: "No auction found in system or auction has been closed", add percept *noAuction* – bidding process has been terminated.
8. Print message: "I submitted the last offer, no action is required this time". Return to the internal action (run the bidding process once again).
9. Print message: "Product's price exceeds the private product's estimation value", add percept *noAuction* – bidding process is terminated.

Smart Agent Implementation

The agent which is presented in this section uses English auction protocol. The main difference between *smart agent* and *dummy agent* is that the former is able to determine price dynamically according to carefully selected tactics. All tactics have been based on decision functions proposed in [Faratin et al., 1998].

The smart agent's business logic includes the following tactics for dynamic price determination:

a) *remaining time* tactic:

$$f_{rt}(t) = \alpha_{rt(t)} \cdot p_r \tag{1}$$

where:

f_{rt} – calculated product price according to remaining time tactic,

t – current time,

p_r – private value of the product (reservation price),

$\alpha_{rt}(t)$ – remaining time tactic coefficient, which is calculated according to formula presented below.

$$\alpha_{rt(t)} = k_{rt} + (1 - k_{rt}) \cdot \left(\frac{t}{t_{max}}\right)^{\frac{1}{\beta_{rt}}} \tag{2}$$

where:

k_{rt} – proportion between initial (p_i) and reservation price (p_r); the initial price is value the agent wants to start bidding process with,

t_{max} – time when the last active action in the English protocol will be terminated,

β_{rt} – adjust shape of the price's curve, it belongs to positive real numbers.

b) *remaining auction* tactic:

$$f_{ra}(t) = \alpha_{ra}(t) \cdot p_r \tag{3}$$

where:

f_{ra} – calculated product price according to remaining auction tactic,

t – current time,

p_r – private value of the product (reservation price),

$\alpha_{ra}(t)$ – remaining auction tactic coefficient, which is calculated according to the formula presented below

$$\alpha_{rt(t)} = k_{ra} + (1 - k_{ra}) \cdot e^{\frac{|L(t)|}{\beta_{ra}}} \tag{4}$$

where:

k_{ra} – proportion between initial (p_i) and reservation price (p_r),

L(t) – number of the open auctions at this time,

β_{rt} – adjust shape of the price's curve, it belongs to positive real numbers.

c) *desire of product purchase* tactic:

$$f_{ba}(t) = \alpha_{ba}(t) \cdot (p_r - \omega(t)) \tag{5}$$

where:

f_{ba} – calculated product price according to desire of purchase product tactic,

t – current time,

$\alpha_{ba}(t)$ – desire of purchase product tactic coefficient,

p_r – private value of the product (reservation price),

ω(t) – minimum purchase price.

$$\alpha_{ba(t)} = k_{ba} + (1 - k_{ba}) \cdot \left(\frac{t}{t_{max}}\right)^{\frac{1}{\beta_{ba}}} \tag{6}$$

where:

k_{ba} – proportion between initial (p_i) and reservation price (p_r),

t_{max} – time when the last active action in the English protocol will be terminated,

β_{ba} – adjust shape of the price's curve, it belongs to positive real numbers.

$$\omega(t) = \frac{1}{|L(t)|} \cdot \left(\sum_{1 \le i \le |L(t)|} \frac{t - \sigma_i}{\eta_i - \sigma_i} \cdot v_i(t)\right) \tag{7}$$

where:

L(t) – number of the open auctions at this time,

σ_i – time when the i-auction started,

η_i – time when the i-auction will end,

$v_i(t)$ – current price of the i-auction.

Algorithm presenting smart agent operation has been shown in the figure 4.

Before agent is initialized, it is possible to modify initial beliefs and rules using AgentSpeak Language. Initial parameters are related to:

- *agent's reaction time* – this value can be deterministic or stochastic. In the prototype, randomly generated values with uniform distribution (5;8) have been set up,
- *parameters for dynamic price calculation tactics* – parameters values can be constant or randomly generated. In the system constant values have been selected,
- *coefficients used in dynamic price calculation tactics* – values that have been set are partially constant and partially random.

After all parameters have been set up, agent starts execution. During initialization stage all coefficients, which are defined in the initial beliefs and rules section are generated and stored in the beliefs base of the agent as constant values. Then agent starts bidding process which is composed of many steps coded with the use of AgentSpeak and Java languages. Bidding is the main plan of the agent. It is executed in the loop until stop conditions are met. Instead of describing every line in detail, the overall idea of agent operation with regard to dynamic pricing will be presented.

The main steps are the following:

- if the agent is going to bid, it waits for a while,
- all prices calculated in the previous iteration are cleared – this is an external action executed by environment,
- all prices are calculated according to tactics – agent's plans for calculation of every tactic's price are almost the same, they take some coefficients as parameters, but all prices are calculated finally in the environment part of the system,
- final bid's value is determined – calculated prices and weights for every tactic are sent to the outside environment to calculate final bid's value,
- auction (for which bid will be submitted) is selected – this is done according to expression that takes under consideration such issues as time remaining to auction's end (auctions which are closer to the end are preferred) and randomly generated value with uniform distribution, which is agent's personal valuation of the product,
- selected auction's ID is saved in agent's beliefs base,
- offer is made with a given auction number and calculated bid value,

- whole process is running in the loop until stop conditions are met.

The stop conditions are stored in the context part of the agent's bidding plan. It is possible that the plan execution will be interrupted with regard to the following reasons:

- bidder has bought the product – bidder successfully executed plan and is not allowed to bid in another auction during current simulation experiment,
- bidder cannot identify any English auction in the system - it means that there is no seller who offers product and therefore it is impossible to bid in the current simulation experiment,
- bidder has not found any auction that is open – if all auctions are closed buyer is not able to submit a bid,
- bidder has already bid – it is prohibited to bid again until the offer is the highest bid.

The business logic that has been described so far has fully been implemented with the use of AgentSpeak language. However there are areas of the prototype developed, where logic programming paradigm needs imperative paradigm support. In these areas Java language has been used for implementation purposes. As a part of smart agent behavior, six external actions have been implemented. They are the following:

- clearing prices calculated during last iteration of the algorithm. This action is quite simple because calculated prices have been stored as beliefs with annotation «percept» denoting the source. Removing them from the beliefs base is done with *clearPercepts* action,
- price calculation according to pricing tactics with given parameters. Calculated price is stored in the beliefs base of the agent,
- bid value calculation according to weights related to every tactic and calculated prices. Final value is stored once again in beliefs base of the agent,
- evaluation and selection of an auction to bid for – evaluation process is performed for every auction which is open. If there is no open auctions, the percept *no(auction)* is returned, which interrupts the bidding process.
  To evaluate every auction value, the following expression is used:

$$f_{val} = rand_{val} \cdot \left(\frac{t-\sigma_i}{\eta_i-\sigma_i}\right) \quad (8)$$

  where:
  $rand_{val}$ – value randomly generated with uniform distribution in a range <0,1)

  other symbols' meaning is the same like in ω(t) expression in formula (7).
  The expression (8) simulates agent's personal valuation of the product, and promotes auctions which are close to the termination. This expression can be easily modified according to the specific bidding strategy planned.

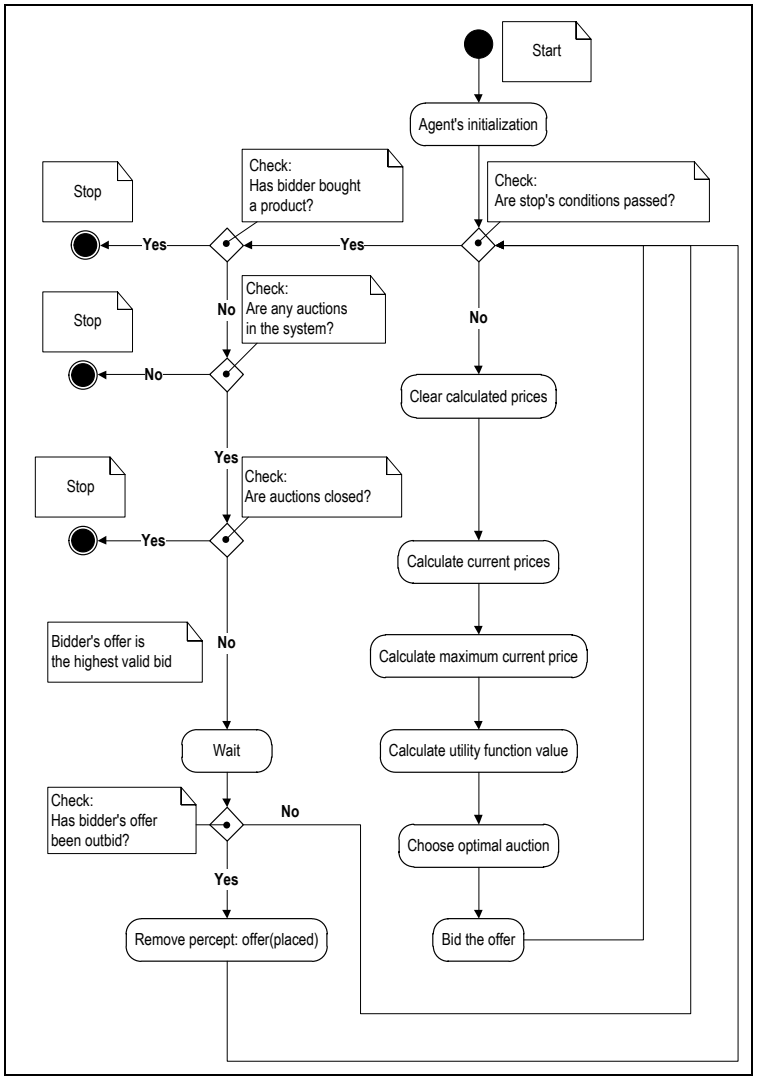


Figure 4. The activity diagram presenting business logic of Smart Agent

- making a bid – as a result of the previous internal and external actions, agent's beliefs base contains two important beliefs: ID of the auction with the highest utility function value and agent offer's value. Both beliefs are bound to the logical variables, which are sent as parameters of the *makeBid* external action. The algorithm of the action is as follows:
    - find auction with given ID,
    - check if the auction is open – if it is not, print message to the console window and terminate action; if it is, go to the next steps,

- check if the last offer does not belongs to the agent – if it does, terminate action and store beliefs *bid(offer)* in agent's beliefs base; if it does not continue operation,
- check if agent's bid value is smaller than or equal to the actual product price – if it is, terminate action and print message to the console window, if it is not, continue operation,
- make a bid,
- save information related to the bidding process in the system.

All auctions and submitted offers are stored in the *ArrayList* data structure, which is created for every auction protocol type. Most important information concerning the bidding process is stored in a text file with semicolon as a delimiter, so data can be further analyzed.

- monitor the auction the agent has bid for - if the agent has successfully bid, making new bid is prohibited until offer has not been outbid. Therefore the action related to monitoring of bidding process of the agent has been implemented.
  If the auction is closed, agent removes belief *bid(offer)* from beliefs base and check if the offer has not been outbid. If it has not, the belief *bought(product)* is added to beliefs base. If the auction is open agent simply checks if the offer is the highest one, and if it is, action ends with no changes made to beliefs base; if it is not, *bid(offer)* belief is removed from beliefs base. Depending on the changes in the beliefs base the agent's activity can be finished, *!bid(auction)* plan can be initialized or monitoring process can be executed once again.

Presented development process concerns the e-marketplace prototype. So far, the main elements of the architecture have been implemented. Of course there is a need for further development but basic business logic has already been implemented. What is more e-marketplace architecture is so flexible that may be easily extended whenever new research objectives will be formulated.

Conclusions and Further Research

The paper presented the first steps of research undertaken in the intersection of MAS and simulation. MABS allows analysis of systems that are too complex to analyze using closed–form techniques. The advantage of using simulation is that it provides us with light where the analytical techniques cast little or none, in our metaphorical search, so we are no longer restricted to working with models which we hope will prove tractable to our analytical tools [Marks, 2006]. After there is an understanding reached thank to simulation, with how the elements of phenomenon of concern work together, it is possible to ask the question of how better to design it.

The final goal the planned research is supposed to achieve is to find the optimal variant of dynamic pricing models business logic that can be implemented in an agent–oriented system supporting management of contracts with suppliers. The analysis process is conducted with the use of MABS, where electronic market plays a role of a workbench for testing different variants of business logic that drive bidding tactics of agents.

Making conclusions from technical perspective leads to the statement, that carefully selected agent's architecture as well as implementation environment seem to be sensible choice. The modeling and implementation process done with BDI architecture and Jason & AgentSpeak tandem have not been so easy. However, having two programming paradigms, object–oriented and logic, in one place gives many advantages. It is possible to easily code the business logic and at the same time make computations when needed.

Further research works will be concerned with equipping agents with intelligent mechanisms such as genetic algorithms in order to improve the performance of agents' bidding tactics and simulation based comparative analysis of these mechanisms.

Bibliography

[Afuah et al., 2002] Afuah, A., Tucci, C., L.: Internet Business Models and Strategies. Text and Cases. McGraw-Hill, (2002).

[Arifovic, 1994] Arifovic, J.: Genetic algorithm learning and the cobweb model. Journal of Economic Dynamics and Control 18, 3–28, (1994).

[Audet et al., 2002] Audet, N., Gravelle, T., Yang, J.: Alternative trading systems: does one shoe fit all?, working paper 2002-33 (Bank of Canada, Ottawa), (2002).

[Bakos, 1991] Bakos, J. J.: A Strategic Analysis of Electronic Marketplaces. MIS Quarterly 15, no. 3 (1991).

[Bellifemine et al., 2005] Bellifemine, F., Bergenti, F., Caire, G., Poggi, A.: JADE — a java agent development framework. In Bordini, R. H., Dastani, M., Dix, J., El Fallah Seghrouchni, A., (eds): Multi-Agent Programming: Languages, Platforms and Applications. No 15 in Multiagent Systems, Artificial Societies, and Simulated Organizations. Springer-Verlag, chapter 3, pp. 69–94, (2005).

[Bordini et al., 2005] Bordini, R. H., Dastani, M., Dix, J., El Fallah Seghrouchni, A., (eds): Multi-Agent Programming: Languages, Platforms and Applications. Number 15 in Multiagent Systems, Artificial Societies, and Simulated Organizations. Springer-Verlag, (2005).

[Bordini et al., 2007] Bordini, R., H., Hubner, J., F., Wooldridge, M.: Programming Multi-Agent Systems in AgentSpeak Using Jason. Wiley Series in Agent Technology. John Wiley & Sons, (2007).

[Bordini et al., 2009] Bordini, R., H., Hubner: Agent-Based Simulation Using BDI Programming in Jason. In: [Weyns et al., 2009] Weyns, D., Uhrmacher, A.M. (eds.): Multi-Agent Systems Simulation and Applications. Computational Analysis, Synthesis, and Design of Dynamic Models Series. CRC Press, Florida (2009).

[Bottazzi et al., 2003] Bottazzi, G., Dosi, G., Rebesco, I.: Institutional architectures and behavioural ecologies in the dynamics of financial markets: a preliminary investigation", Technical Report, Laboratory of Economics and Management, Sant' Anna School of Advanced Studies, Pisa, Italy, (2003).

[Bullard et al., 1999] Bullard, J., Duffy, J.: Using genetic algorithms to model the evolution of heterogeneous beliefs. Computational Economics 13 (1), 41–60, (1999).

[Davidsson , 2002] Davidsson, P.: Agent based social simulation: a computer science view, J. Artif. Soc. Social Simulation, 5, (2002).

[Duffy, 2006] Duffy, J.: Agent-based models and human-subject experiments. in: Tesfatsion, L. Judd, K. L. (Eds): Handbook of computational economics. Volume 2 –Agent-based Computational Economics, North Holland, (2006)

[Faratin et al., 1998] Faratin P., Sierra, C., Jennings, N., R.: Negotiation Decision Functions for Autonomous Agents, In: Journal of Robotics and Autonomous Systems, pp. 159-182., (1998).

[Ferber et al., 2009] Ferber, J., Michel, F., Drogoul A.: Multi-Agent Systems and Simulation: A Survey from the Agent Community's Perspective. In: [Weyns et al., 2009] Weyns, D., Uhrmacher, A.M. (eds.): Multi-Agent Systems Simulation and Applications. Computational Analysis, Synthesis, and Design of Dynamic Models Series. CRC Press, Florida (2009).

[Gjerstad, 2004] Gjerstad, S.: The impact of bargaining pace in double auction dynamics. Department of Economics, University of Arizona, (2004).

[Jakieła, 2006] Jakieła, J.: AROMA – Agentowo zorientowana metodologia Modelowania orgAnizacji. WAEiI, Politechnika Ślaska, Gliwice (2006)

[Jakieła et al., 2009] Jakieła J., Pomianek B.: Agent Orientation as a Toolbox for Organizational Modeling and Performance Improvement. International Book Series "Information Science and Computing", Book 13, Intelligent Information and Engi-neering Systems, INFOS 2009, pp. 113-124, (2009).

[Jakieła, Litwin, Olech, 2010a] Jakieła J., Litwin P., Olech M.: Toward the Reference Model for Agent-based Simulation of Extended Enterprises. In: Setlak, G., Markov, K.: Methods and Instruments of Artificial Intelligence, pp. 34-66, (2010).

[Jakieła, Litwin, Olech, 2010b] Jakieła, J., Litwin, P., Olech, M.: MAS Approach to Business Models Simulations: Supply Chain Management Case Study. In: KES AMSTA-2010, Jędrzejowicz, P. Nguyen, N., T., Howlett, R., Lakhmi, C. J., (Eds.), Part II, LNAI 6071, pp. 32-41, Springer-Verlag, Berlin Heidelberg, (2010).

[Jakieła, Litwin, Olech, 2011] Jakieła J., Litwin P., Olech M.: Prototyp platformy symulacji wieloagentowej rozszerzonych przedsiębiorstw. Studia Informatica, vol. 32, Number 2B (97), pp. 9-23, Gliwice (2011).

[Jakieła, Litwin, Olech, 2012] Jakieła J., Litwin P., Olech M.: Multi-Agent Based Simulation as a Supply Chain Analysis Workbench. Issues 6, Springer-Verlag, Berlin Heidelberg, Transactions on Computational Collective Intelligence, LNCS, vol. 7190, pp. 84-104, (2012).

[LeBaron, 2006] LeBaron, B.: Agent-based computational finance in: Tesfatsion, L., Judd, K. L. (Eds): Handbook of computational economics. Volume 2 –Agent-based Computational Economics, North Holland, (2006).

[Luck et al., 2003] Luck, M., McBurney, P., Preist, C.: Agent technology: enabling next generation computing. A roadmap for agent based computing, (2003), www.agentlink.org.

[MacKie-Mason et al., 2006] MacKie-Mason, J. K., Wellman, M. P.: Automated Markets and Trading Agents In: Tesfatsion, L., Judd, K. L. (Eds): Handbook of computational economics. Volume 2 – Agent-based Computational Economics, North Holland, pp. 584-634 (2006).

[Marks, 2006] Marks R.: Market Design Using Agent-Based Models. in: Tesfatsion, L., Judd, K.L. (Eds) : Handbook of computational economics. Volume 2 –Agent-based Computational Economics, North Holland, (2006).

[Milgrom, 2004] Milgrom P.: Putting Auction Theory to Work. Cambridge University Press, (2004).

[North et al., 2007] North, M., J., Macal, C.M.: Managing Business Complexity. Discovering Strategic Solutions with Agent-Based Modeling and Simulation. Oxford University Press (2007).

[Parunak et al., 1998] Parunak, H., V., D., Savit, R., Riolo, R., L.: Agent-based modeling vs. equation-based modeling: A case study and users' guide. In J. S. Schiman, R. Conte, and N. Gilbert, editors, Proceedings of the 1st Workshop on Modeling Agent Based Systems, MABS'98, volume 1534 of LNAI. Springer-Verlag, (1998).

[Rao et al., 1992] Rao, A., S., Georgeff, M., P.: An abstract architecture for rational agents. In: Nebel, B., Rich, C., Swartout, W., R. (Eds), Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning (KR'92), pp. 439–449. Morgan Kaufmann Publishers, (1992).

[Turban et al., 2011] Turban, E., King, D., Viehland, D., Lee, J.: Electronic commerce. A managerial perspective. Prentice-Hall, (2011).

[Weyns et al., 2009] Weyns, D., Uhrmacher, A.M. (eds.): Multi-Agent Systems Simulation and Applications. Computational Analysis, Synthesis, and Design of Dynamic Models Series. CRC Press, Florida (2009).

[Zambonelli et al., 2002] Zambonelli, F., Parunak, H., V., D.: From design to intention: signs of a revolution. In Proceedings of the first international joint conference on Autonomous agents and multiagent systems, pp. 455–456. ACM Press, (2002).

Authors' Information

Jacek Jakieła, Ph.D., Eng. *– Department of Computer Science FMEA RUT; Powstancow Warszawy ave. 12, 35-959 Rzeszow, Poland; e-mail: jjakiela@prz.edu.pl*

Major Fields of Scientific Research: Software Development Methodologies, Agent and Object-Oriented Business Modeling, Internet Enterprises Models, Computational Organization Theory and Multi-Agent Based Simulation of Business Architectures.

Paweł Litwin, Ph.D., Eng. *– Department of Computer Science FMEA RUT; Powstancow Warszawy ave. 12, 35-959 Rzeszow, Poland; e-mail: plitwin@prz.edu.pl*

Major Fields of Scientific Research: Applications of Neural Networks in Mechanics, Computer Simulations, Finite Element Method.

Marcin Olech, M.Phil., Eng. *– Department of Computer Science FMEA RUT; Powstancow Warszawy ave. 12, 35-959 Rzeszow, Poland; e-mail: molech@prz.edu.pl*

Major Fields of Scientific Research: Multi-agent Based Simulations, Artificial Intelligence Applications in Industry.

MULTI-AGENT SYSTEM FOR SIMILARITY SEARCH IN STRING SETS

Katarzyna Harężlak, Michał Sala

***Abstract**: The aim of the paper is to present the assumptions and the architecture of the system for searching similarity in string sets. During the research all the required steps of a procedure of text documents processing which includes text extraction, pruning, stemming and lemmatization were analysed. Models of a text documents' description and the method of creating a vector of features were developed as well. This vector consists, inter alia, of chosen words and the number of their occurrences. The process of the text analysis is supported by a set of various dictionaries. These are Stop-words, Domain and Lemma dictionaries and all of them were considered in the context of the Polish language. Because the Lemma dictionary is supposed to consist of many entries, the efficient method of its access optimisation was elaborated. Various measures used for calculating degree of a text documents similarity were studied too. Moreover, the method for determining the quality of user queries and text documents adjustment were proposed.*

The system was realized in accordance with the idea of multi-agent systems. Its functionality is ensured by the set of agents acting on the basis of separate threads. In the research, tests of the system work efficiency were also performed.

***Keywords**: agent systems, text similarity search*

***ACM Classification Keywords**: I.7 Document And Text Processing*

Introduction

Knowledge is an inseparable, continuously extended, element of the modern life. Fast, regarding various areas, science development results in providing us with many books, articles and web information sources which, in the era of widespread use of computers and global networks, makes access to knowledge unlimited. For this reason, while searching for information to understand a given issue better, many sources should be analysed. Making this process useful and effective entails the need of creating methods ensuring fast access to particular information.

The simplest solutions, based on metadata or patterns searching, have two disadvantages. The result of their action consists of too much information, which does not match a given pattern – in the worst case scenario potentially valuable documents can not include words defined in the search criteria. Next drawback is a possibility of important information loss – this situation can take place when a searched document is determined by too small or inappropriate set of metadata. What is more, two different papers indexed by the same set of words can be classified as similar documents whereas in reality refer to different areas.

More advanced systems use more precise analysis including text comparison and classification [Bollacker, 1998, Aggarwal 2001]. In this case the notion of documents' similarity, relying on defining correspondence degree between documents being compared, has been introduced. Owing to this rate, the problems of obtaining too large result set and possibility of information loss can be reduced. The text semantic similarity is mainly determined on the basis of documents' contents analysis and comparison. Considering this idea on the high abstraction level two main methods can be distinguished. The first one uses text matching to point out the same part of documents. In this case the similarity is determined by the number and quantity of repeated contents. In the second method sets of features, describing text content, are generated, which are subsequently used by functions calculating the distance between two analyzed documents.

Documents written in the natural language, from the computer analysis point of view, feature high redundancy of information. Variety of declensions, words or punctuation marks are useless for methods used for text analysis. Therefore, for their need, digital representation of document's description allowing for mathematical analysis is required. One of the simplest examples is a unigram model, which assumes that documents are described by vectors whose values represent the existence of given words. The effective vector generation, for a particular document, requires carrying out preliminary processing of its content. This activities are responsible for removing unimportant text elements, which have no influence on a text comparison and changing all declensions to basic forms of particular words [Strömbäck, 2005, Dąbrowski, 1978,Smirnov, 2008].

The aim of this paper is to present the assumptions and the architecture of the system for searching similarity in string sets. The system was developed in accordance with the idea of multi-agent systems. Its functionality is ensured by the set of agents acting on the basis of separate threads.

The Multi-Agent Systems

Multi-Agent System – MAS [Bigus, 2001, Wooldridge, 2009] is a system comprising of many intelligent agents which collaborate to solve a given goal. Agents usually are defined as units making autonomous decisions on behalf of a user or a superior system to realize assigned them functions. Their activity rely on continuous observation of objects belonging to a specific environment and choosing, on the basis of the states of these objects, appropriate actions influencing the environment.

Agents are characterized, dependently on their applications, by various features. From this paper point of view these are reactivity, autonomy, cooperation and coordination.

It means that agents have to observe their environments and respond to occurring changes, individually decide which activities should be taken and exchange needed resources to complete chosen actions and coordinate their execution.

There are many frameworks supporting agent system building but in case of presented research C# language and .NET class were used.

The System Architecture

The vision of the system, with taking idea of agent system into account, is presented in the figure 1a. It consists of three elements: the **Environment**, **Superior Layer** and **Agents**, divided into two categories. The first category comprises of Processing Agents, which produce data used by the second group in the process of documents comparison. They observe an environment detecting presence of source documents to be analysed. In case of existence of such document, agent removes it from the environment and prepares for further analysis. After that document, as a processed one, is returned to the environment. This is an impulse activating agents from the second category. They assess a documents in terms of meeting search criteria and, in case of success, add document to the result set. Regardless of reacting to environment ‘s changes, agents can communicate between one another to synchronize their activities.

The Superior Layer is the application, which defines user interface and starts, on user’s request, the process of documents’ analysis. This operation entails defining source of documents to analyse and providing a set of search conditions defined by a user. The above mentioned elements, in conjunction with processed documents, constitute the last component in the system vision - **the Environment**. The illustrative model of the main functionality of the system is presented in the figure 1b.

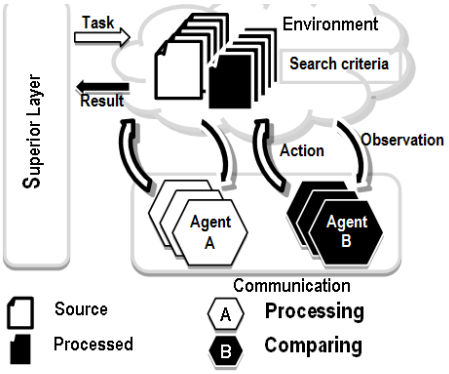


Fig.1a. The vision of the system

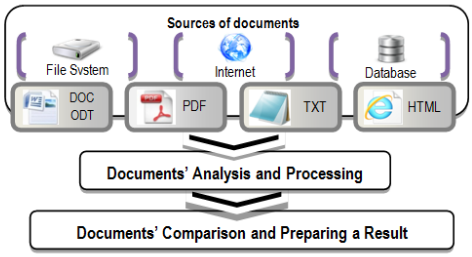


Fig.1b. The illustrative model of the main functionality of the system

The System Modules

The system was divided into few modules (Fig. 2). The **Main** one controls a process of starting the system, its work and results presentation. This element includes implementation of the user interface enabling access to all functions of the system. Its duty is to create agents' threats and to assign them appropriate objects gathering data on agents' states. In the Main module an instance of the environment class is created, which is passed to all newly created agents. In addition, among functionality of this module can be found:

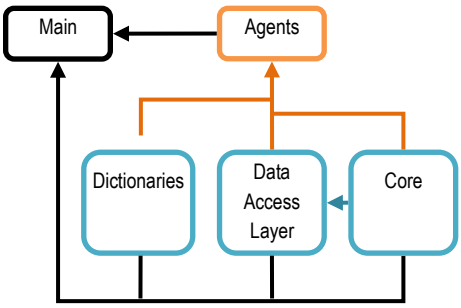


Fig. 2. The system modules

- loading and modifying domain dictionaries,
- changing user requests to lemma strings
- loading documents from particular sources (file system, Internet), database or xml files,
- saving processed documents to the database or xml files.

The **Agents** module includes implementation of agents processing and comparing documents. Processing Agents comprise logic facilitating such operation like pruning, lemmatization and counting numbers of lemma occurrences, searching and counting patterns in an analysed document. Agents of the second type, apart from comparing processed documents, are responsible for determining a quality of documents and search criteria matching.

The **Core**, **Dictionaries** and **Data Access Layer** constitute auxiliary modules responsible respectively for delivering the basic structures and functionality, as well as providing access to the set of dictionaries and sources of documents to be processed. The schema of modules collaboration is presented in the figure 3a and 3b.

Documents processing

During the research presented in the paper for documents analysis the n-gram model was used [Cavnar 1994, Palus, 2011]. In accordance with this model numbers of occurrences of given words sequences are stored in form of vectors of features representing characteristics of documents. Features in vectors are represented by a collection of keys and values corresponding to them.

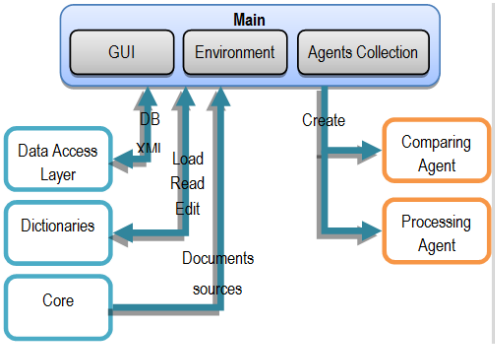


Fig.3a. The Main module and its dependencies

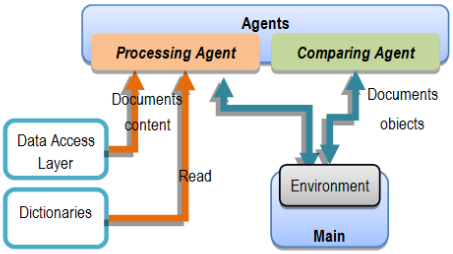


Fig.3b. Agents collaboration

Defining vectors of features, requires going through document contents with simultaneous calculating number of occurrences of particular words or phrases. However this process is performed in documents whose contents have been transformed by operations of lemmatization and pruning earlier [Borycki, 2002, Nguyen, 2009]. These are procedures, in which word's declinations are replaced by a common term, which, dependently on a given method, is a lemma or a core of a word. A lemma is basic form of a word, while core does not have to be proper word and is developed as a result of a inflectional transformations. If it is taken into account, that word, being result of the lemmatization process, is the same word, which is searched in the analyzed document, defining a vector of features can be performed in the same step.

In both procedures, three types of dictionaries – the **Stop-words**, **Domain** and **Lemma** one – play a significant role. First of them is used to remove from a document all unimportant elements like "but", "and", "why" or "whatever". Content of the Domain dictionary allows for finding common contents of documents and classifying them in terms

of belonging to a particular field. This dictionary is divided into as many files as domain is to be analysed and can be complemented by a user, also with usage of the system suggestions, based on current analysis of documents. For defining last of the aforementioned dictionaries – the Lemma one, including lemma of polish words – text form of content of the Morfologik dictionary, was used [Morfologik, 2011].

All dictionaries are stored in the file system and during the system start they are loaded into appropriate structures. In case of the Stop-words dictionary it is a simple list of elements, whereas structures of the Domain and the Lemma dictionaries are divided into sections. Units of the first of these two dictionaries represent particular fields, while division of the lemma dictionary is based on the common prefix of a word.

Searching user criteria refers to the problem of searching patterns in character strings. There are many algorithms which can be used to solve this problem. Among them well known algorithms like *Naive*, *Boyer-Moore* or *Knutt-Morris-Pratt* one can be mentioned [Cormen, 2001]. Two last algorithms are effective solutions for text pattern searching, but they are characterized by one feature, which can be considered as a disadvantage. It is an ability to match at most one sequence in one run and, as a consequence, there is the lack of possibility of searching for many patterns concurrently. To remove this inconvenience the solution relaying on creating an index, which points to, for each searched sequence, all places of occurrences of its first elements, was proposed. Navigating through the index entries enables finding particular patterns quickly. However, preparing such index requires scanning a document to find proper positions. It can be done, like in the case of counting words, during lemmatization phrase. The idea of the index is presented in the figure 4.

Calculating number of occurrences of a phrase is more complicated task then in case of a single word. At the beginning, a set of all possible sequences of words belonging to search criteria must be determined. Such sequences can consists of few words of length between 2 and n. Moreover, in case of phrases existing in Polish language, the order of words doesn't affect meaning of a phrase. For this reason all possible sequences without repetition of size between *2* and *n* from set of size *k* have to be taken into consideration.

In the presented research described problem was solved by a use of an algorithm building appropriate tree, in which all but one (root) nodes represent words and each path in the tree, with start point in the root element, constitutes one of a pattern sequences.

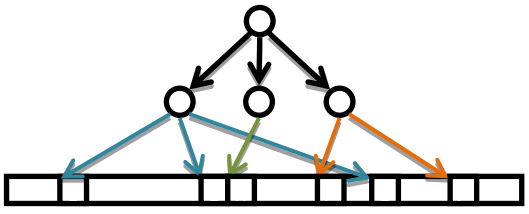


Fig. 4. The example of an index pointing occurrences of first elements of sequences

Searching phrases in an analyses document begins with finding first occurrence of a first item of the aforementioned index. Then subsequent node in the sequences tree is compared with the consecutive word in a document. Presence of a difference means that pattern was not found in that place of a document. Such operation is repeated for all sequences beginning with the same word of a pattern and for all words beginning other pattern sequences.

Documents Comparison

One of the main goals of the presented system is to deliver a tool for comparing and assessing similarity of text documents. As it was described in the previous sections, the system element responsible for providing such functionality is, equipped with appropriate structures and logic, the Comparing Agent. For determining documents matching it uses vectors of features defined earlier and some distance and proximity measures like Hamming or Euclidean one [Deza, 2009]. However, these measure had to be modified for the purpose of this research, because of an existence of one disadvantage – the difficulty to transform values returned by them to a percentage scale, which reflects a documents conformance better. In case of similarity measures it can be difficult to determine a value, which reflects complete documents matching. Likewise, using distance measures, a values describing maximal difference between documents is not known.

Obtaining results in a percentage scale requires introducing some changes to basic formulas of previously mentioned measures. Assuming following symbols:

P – documents distance measure in percentage scale,

n – number of features,

u, v – vectors of features of two documents

the modified Hamming measure is defined as follows:

$$P = \frac{1}{n}\sum_{k=1}^{n}\frac{|u_k - v_k|}{\max(u_k, v_k)}100\% \tag{1}$$

According to this formula the percentage ratio of dissimilarity, for each pair of documents, is calculated. To represent it in a form of a function determining documents similarity in direct way, next modification should be applied (Equ. 2)

$$similarity = \left(1 - \frac{1}{n}\sum_{k=1}^{n}\frac{|u_k - v_k|}{\max(u_k, v_k)}\right)100\% \tag{2}$$

In order to improve the quality of documents comparison, features characterizing documents are differentiated by various weights, to strengthen meaning features representing search criteria. What is more weights of phrases are higher, because their occurrence is more unique than simple words. The final formula for calculating similarity is defined by the equation 3:

$$similarity = \left(1 - \frac{1}{m}\sum_{k=1}^{n}\frac{w_k \left| u_k - v_k \right|}{\max(u_k, v_k)}\right)100\% \tag{3}$$

where:

w – vector weights of features,

w_k – weight of k feature,

$m = \sum_{k=1}^{n} w_k$ – sum of weights.

The rule of weighted features was applied to other distance and proximity measures, and in form of the following equations, were embedded into the Comparing Agent logic :

1. weighted Hamming distance measure

$$P(u,v) = \sum_{k=1}^{n} w_k . \left| u_k - v_k \right| \tag{4}$$

2. weighted Euclidean distance measure

$$P(u,v) = \sqrt{\sum_{k=1}^{n} w_k . \left| u_k - v_k \right|} \tag{5}$$

3. proximity measure based on number of occurrences of search words

$$P(u,v) = \sum_{k=1}^{n} w_k u_{k.} v_k \tag{6}$$

4. proximity measure based on values of features

$$P(u,v) = \sum_{k=1}^{n} w_k . \min(u_{k.} v_k) \tag{7}$$

5. proximity measure based on values of features, taking into account also the documents characterized by low factors

$$P(u,v) = \sum_{k=1}^{n} w_k . \frac{\min(u_k, v_k)}{\max(N(u), N(v))}, \qquad where \;\; N(x) = \sum_{i=1}^{n} w_i . x_i \tag{8}$$

The Result Quality

The result quality is defined as a value determining, for a given document, the degree of matching the search criteria. It is calculated on the basis of the same vectors of features, which are used in the process of comparing documents. Using weighted Hamming measure and assuming that user condition vector is set to zero, unambiguous values of a quality of documents conformance can be obtained. The higher the value the better the criteria met. A percentage representation of a result requires its linear scaling in the range of 0% and 100% (Equ. 10).

$$quality = \sum_{k=1}^{r} w'_k \left| 0 - v'_k \right| = \sum_{k=1}^{r} w'_k \, u'_k \tag{9}$$

$$quality_{\,percentage}(i) = \frac{qualilty\,(i)}{\max\limits_{j}\,(quality\,(j))} 100\% \tag{10}$$

where:

r – a number of features belonging to a search criteria,

u' – a vector consisting of document's features corresponding to a search criteria,

w' – weights for u' – a vector of features,

i, j – indexes identifying particular documents.

Determining quality of documents' matching is performed by the Comparing Agent during process of their analysis, but percentage values is generated at the end, after processing all documents, because the highest value of similarity must be calculated first.

The Result Presentation

The effect of the system work is a set of many, compared in pairs, documents with associated quality matching rates. A text form of a result including such large amount of data is difficult to analyse, but the same refers to a graphical representation, which provides a complicated network of linked documents (Fig 5a). This is why few mechanisms were developed to facilitate presentation of obtained results (Fig. 5b):

1. presenting links only for a chosen document,
2. presenting all links with accentuating edges connected to a given node,
3. providing possibility of limiting links description,
4. providing possibility of filtering network on the basis of the lowest similarity rates.

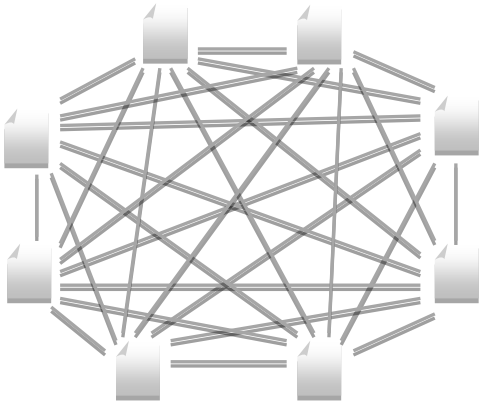


Fig. 5a. Sample network of linked documents

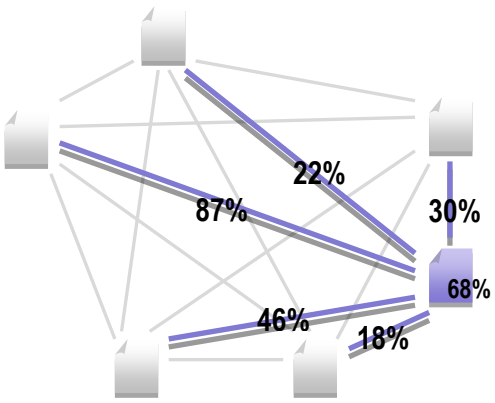


Fig. 5b. The same network with developed mechanisms aplied

Additionally, in the system, the functionality of presenting details regarding similarity rate, after choosing a given link, was developed as well.

Experiments

The developed system was tested in terms of checking how some parameters or methods, can influence its work efficiency. The set of examined elements included the method of storing the Lemma dictionary, the length of searched sequences and methods used for documents comparison.

During the research it was assumed that the Lemma dictionary can be characterized by a large size, so its searching can be crucial for documents processing. Therefore, the decision to divide content of this dictionary into smaller sections was made. However a way of the division and size of a unit had to be examined. As a condition for dictionary splitting, word's prefix was utilized. Performed tests were to show how long such prefix should be. A length of a prefix was determined experimentally by analysis of response time of searching example words.

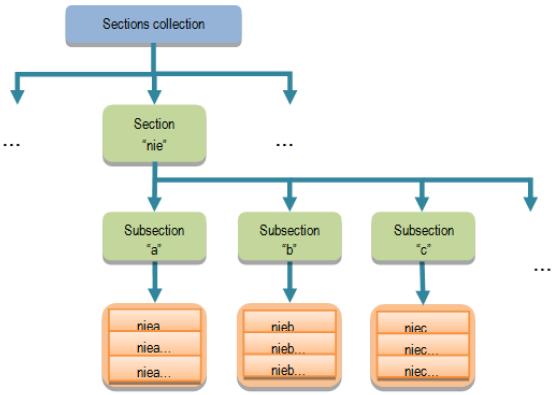


Fig.6. The structure of the lemma dictionary

Obtained findings proved that, in case of Polish language, prefixes of the length of three characters gave the best result but in same kinds of words four characters were needed to reduce search time (Table 1). To improve these outcomes in case of some lemma dictionary sections, additional level, consisting of words having the same character after the prefix, was introduced (Fig. 6).

Table 1. Sample results of experiments – time of lemma dictionary access in milliseconds

Length of the	**1**	**2**	**3**	**4**	Results for prefix consisting of three characters and additional level for words staring with prefix “nie”	
cytryna	55	1	0	2		0
nieznajomy	305	260	263	18		14
tttratwa	23	0	1	4		1

During subsequent tests, influence of the length of a sequence - representing a feature describing document - on time of its processing, was studied. Table 2 includes sample results obtained for 30 documents returned by Google search engine for sequences of length from 1 to 5. It can be noticed that the time of documents processing has a slight upward trend, so the conclusion saying that the length of a feature sequence does not have great impact on total processing time can be drawn.

Table 2. Influence of a sequence length on time of document’s processing

Sequence length - n	**1**	**2**	**3**	**4**	**5**
Test 1, time [ms]	81266	81372	80289	82458	90969
Test 2, time [ms]	82534	83296	82773	82679	95545

The last examined aspect concerned an impact of a chosen comparing method on efficiency of system work. Once more sets of documents obtained from the Internet were used. These documents were analysed by two Processing and three Comparing Agents. Results of two sample tests, consisting of 30 and 10 documents respectively, are shown in the table 3.

Table 3. Influence of a comparing method on time of document’s processing

Comparing method	**4**	**5**	**6**	**7**	**8**	**1**
Test 1, time [ms]	59459	51147	53591	54005	48343	51533
Test 2, time [ms]	19221	20581	18634	18373	19519	19174

Numbers of methods visible in the table represent numbers of equations presented in the chapter Documents Comparison. Analysing obtained values, it is difficult to point out the best, in terms of processing efficiency, method. Within a given test, times are comparable, but between tests, in the regard to particular methods, they become contradictory. Therefore, providing user with a possibility to choose the comparing method seems to be good idea (fig. 7).

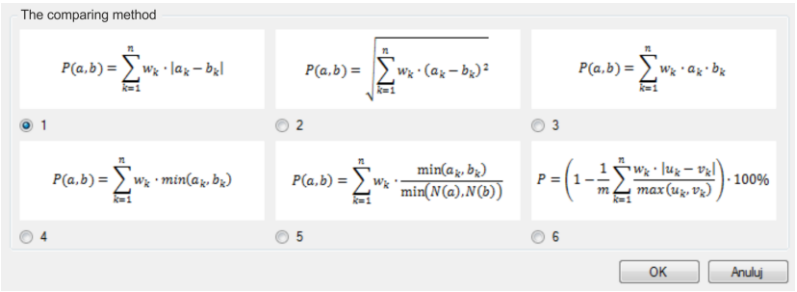


Fig. 7. The system window allowing for defining the search environment

Conclusion

The main goal of the research was to build the system, which searches documents in defined sources, processes them and shows existing similarities. Additional assumption regarding the system architecture, to be in accordance with idea of the multi-agent systems, was made.

After the analysis of the text processing issue, the document representation based on a vector of features was chosen. Documents' matching, in such case, relies on comparing values of particular features. Achieving this functionality was possible owing to the Lemma dictionary usage. Because this element was expected to be the crucial one in terms of efficiency of the system work, some experiments, examining it, were performed. They made allowances for elaborating solutions optimizing the Lemma dictionary search. These solutions, including the index of sequences of search criteria elements and the Lemma dictionary structure, improved the efficiency of this element analysis.

Experiments performed during the research showed that both length of a search sequence and a method chosen for documents' comparison do not have significant impact on the efficiency of this process.

A few other requirements were also raised with respect to the project, such as the possibility of saving results and their graphic presentation. There were secondary features but also important for the final effect. Their fulfillment allowed to make the application not only universal but also user-friendly.

Bibliography

[Aggarwal 2001] C. C. Aggarwal, P. S. Yu. On Effective Conceptual Indexing and Similarity Search in Text Data in Proceeding ICDM '01 Proceedings of the 2001 IEEE International Conference on Data

[Bigus, 2001] J.P. Bigus, J. Bigus. Constructing Intelligent Agents Using Java, 2nd edn. John Wiley & Sons, Inc., New York, NY, USA. 2001

[Bollacker, 1998] K. D. Bollacker, S. Lawrence, C. L. Giles. CiteSeer: An Autonomous Web Agent for Automatic Retrieval and Identification of Interesting Publications. In: Proceeding: AGENTS '98 Proceedings of the second international conference on Autonomous agent, 1998.

[Borycki, 2002] Ł. Borycki, P. Sołdacki. Automatic text classification (Automatyczna klasyfikacja tekstów). 2002, http://www.zsi.pwr.wroc.pl/zsi/missi2002/pdf/s504.pdf, 2011

[Cavnar 1994] B.W. Cavnar, J. M. Trenkle. N-gram-based text categorization. In Proceedings of SDAIR, 1994

[Cormen, 2001] H. Cormen, R. L. Rivest, C. E. Leiserson. Introduction To Algorithms, MIT Press, 200

[Dąbrowski, 1978] M. Dąbrowski, K. Laus-Mączyńska. Methods of Information Search and Classification (Metody wyszukiwania i klasyfikacji informacji), WNT, Warszawa 1978

[Deza, 2009] M. M. Deza, E. Deza Encyclopedia of Distances. Springer-Verlag Berlin Heidelberg 2009

[Morfologik, 2011] Morfologik, http://morfologik.blogspot.com/, 2011

[Nguyen 2009] L. T. Nguyen. Static Index Pruning for Information Retrieval Systems: A Posting-Based Approach. In Proceedings of 7th Workshop on Large-Scale Distributed Systems for Information Retrieval (LSDS-IR'09)

[Palus, 2011] A. Pauls D. Klein. Faster and Smaller N-Gram Language Model. in Proceeding HLT '11 Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, 2011.

[Smirnov, 2008] I. Smirnov. Overview of Stemming Algorithms. DePaul University, http://the-smirnovs.org/info/stemming.pdf, 2011

[Strömbäck, 2005] P. Strömbäck. The Impact of Lemmatization in Word Alignment, Department of Linguistics and PhilologySpråkteknologiprogrammet 2005

[Strömbäck, 2005] P. Strömbäck. The Impact of Lemmatization in Word Alignment, Department of Linguistics and Philology Språkteknologiprogrammet 2005

Authors' Information

Katarzyna Harężlak *– Silesian University of Technology;*
e-mail: katarzyna.harezlak@polsl.pl

Major Fields of Scientific Research: Distributed and Mobile Databases, Software Engineering, Agent Systems, Social Networks

Michał Sala *– Silesian University of Technology;*
e-mail: michal.sala@live.com

Major Fields of Scientific Research: Text searching, Agent Systems

ARTIFICIAL INTELLIGENCE IN MODELING AND SIMULATION

DECOMPOSITION METHODS FOR LARGE-SCALE TSP

Roman Bazylevych, Marek Pałasiński, Roman Kutelmakh, Bohdan Kuz, Lubov Bazylevych

***Abstract**: Decomposition methods for solving large-scale Traveling Salesman Problem (TSP) are presented. Three approaches are proposed: macromodeling for clustered TSP as well as extending and "ring" methods for arbitrary points' distribution. Four stages of the problem solving include partitioning of the input set of points into small subsets, finding the partial high quality solutions in the subsets, merging the partial solutions into the complete initial solution and optimizing the final solution. Experimental investigations as well as the comparative analysis of the results and their effectiveness estimation in terms of quality and running time were conducted. The suggested approaches provide substantial reduction in the running time in comparison with the existing heuristic algorithms. The quality loss is small. The problem instances up to 200,000 points were investigated. The TSP is extensively applied in transportation systems analysis, printed circuit boards, VLSI, SoC and NoC computer-aided design, testing and manufacturing, laser cutting of plastics and metals, protein structure research, continuous line drawings, X-ray crystallography as well as in number of other fields.*

***Keywords**: traveling salesman problem, combinatorial NP-hard problems, decomposition, large-scale.*

***ACM Classification Keywords**: G.2.1 Combinatorics - Combinatorial algorithms; I.2.8 Problem Solving, Control Methods, and Search - Heuristic methods.*

Introduction

The Traveling Salesman Problem (TSP) belongs to the class of intractable combinatorial ones (NP-hard). It consists in finding the shortest route through the set of points provided that each point is visited one time. The complexity of the TSP is *O(n!)*. The largest

problem for which the optimal solution was found consists of 85900 points [Applegate, 2009]. The computations, though, required about 136 years of CPU time. The proposed approaches are developed for the large-scale TSPs to receive the solutions close to optimal in a reasonable time. The problem under discussion is tightly connected with such areas as logistics, scheduling, robot management systems, analysis and synthesis of chemical structures, continuous line drawings, integrated circuit design, manufacturing, etc.

The existing effective heuristic approaches are characterized by the time complexity not less than $O(n^2)$. The most advanced software application to date that allows finding the optimal route is Concorde [Concorde]. The route close to optimal may be obtained through the Lin-Kernighan-Helsgaun (LKH) algorithm [Helsgaun, 2000, 2006]. Efficient way to solve the large-scale TSP is to use the data decomposition approaches [Reinelt, 1994, Rohe, 1997], [Yll Haxhimusa, 2009]. Some decomposition methods are proposed in [Bazylevych, 2007, 2008, 2009, 2011].

Decomposition approaches

The developed decomposition algorithms for the large-scale TSP solving consist of four main stages:

1. Input data set decomposition into subsets.
2. Finding the partial solutions for subsets.
3. Merging the partial solutions into the one complete initial solution.
4. Applying the optimization algorithms to the initial solution.

Data set decomposition allows splitting a huge problem into the set of the smaller ones. Small problems may be easily solved using the known approaches that guarantee getting high quality results. Each subset has a limited number of points. This number depends on the compromise between quality and runtime. The TSP for every cluster is solved separately by the chosen basic algorithm which provides a high quality solution. The exact algorithms, such as branch-and-bound, could be used at this stage.

For the clustered TSP, macromodeling approach is proposed in [Bazylevych, 2007]. Every cluster is approximated by the macromodel as one point. The macroroute for a set of such points as well as the routes (partial solutions) for all clusters are found by the chosen basic algorithm. All partial solutions are concatenated at the next stage with the creation of the complete initial solution.

For the arbitrary TSP, we split a huge problem into some set of the smaller ones, which have a limited number of points (subsets), and then find the partial solutions for

them [Bazylevych, 2008, 2009, 2011]. These processes could be executed in parallel. As a result of this stage, the number of partial tours is obtained, each tour per subset.

At the third stage we merge the partial tours into one complete tour – the initial solution. Two approaches we proposed. As for the first one, merging process consists in extending the partial TSP solutions [Bazylevych, 2008, 2009]. As for the second approach, merging is performed over the regions called the "rings" that are formed from the border points of given and adjacent subsets [Bazylevych, 2011]. The result of this stage is the tour which passes through all the points. The complete initial solution is also received by independently merging the tour pieces for every "ring".

The quality of the complete initial solution is improved by applying some specially developed optimization methods [Bazylevych, 2007, 2008, 2009, 2011].

Methods for initial solution

For receiving the initial solution we developed such methods:

1. Macromodeling for clustered TSP.
2. The extension method.
3. The "Ring" method.

Macromodeling for clustered TSP

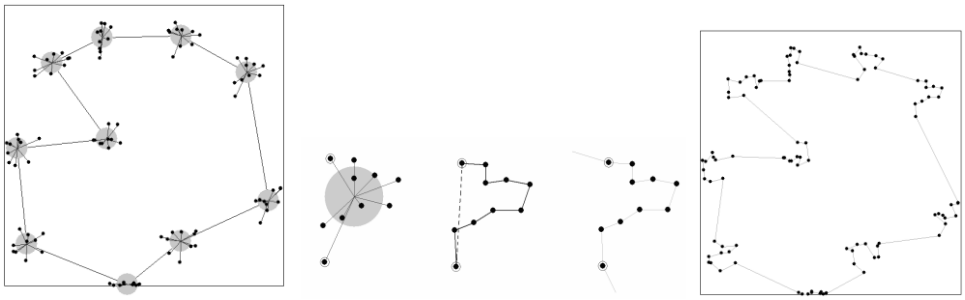
The main steps of the method [Bazylevych, 2007] developed for the clustered TSP are (Figure 1):

a) Macromodeling by finding the minimal length macroroute which passes through every cluster only once. Every cluster is approximated by one point (Figure 1a).

b) Micromodeling by finding the partial solutions in every cluster. This process consists of two steps:
   - finding the shortest route between the adjacent clusters and setting the border points for every cluster;
   - finding the shortest route between the border points for every cluster (Figure 1b).

c) Finding the complete initial solution by concatenating the routes between the clusters with partial routes for all clusters (Figure 1c).

d) Optimizing the solution.



a) Macroroute b) Internal subroute in the cluster c) Complete initial solution

Fig. 1. Macromodeling for Clustered TSP

The extension method

The main steps of the method [Bazylevych, 2008, 2009] developed for the arbitrary TSP are (Figure 2):

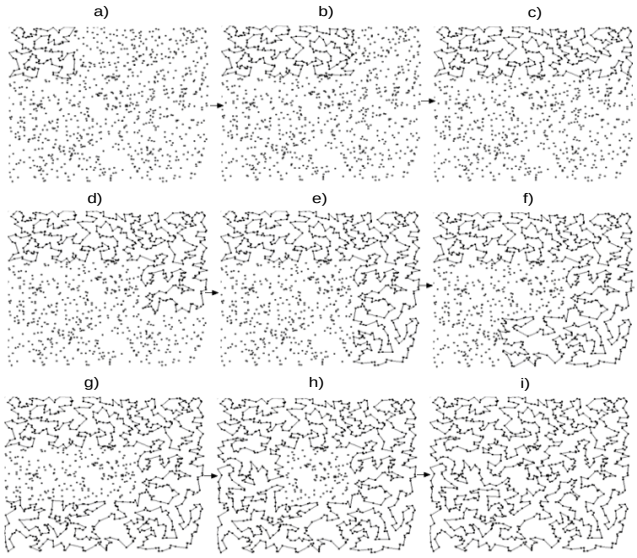


Fig. 2. Example of extending the partial solutions in spiral inner order

a) Choosing a small geometrical area from the whole problem surface in which it would be possible to obtain the high quality TSP solution. It could be the area from the corner (arbitrary) or from the center of the surface.

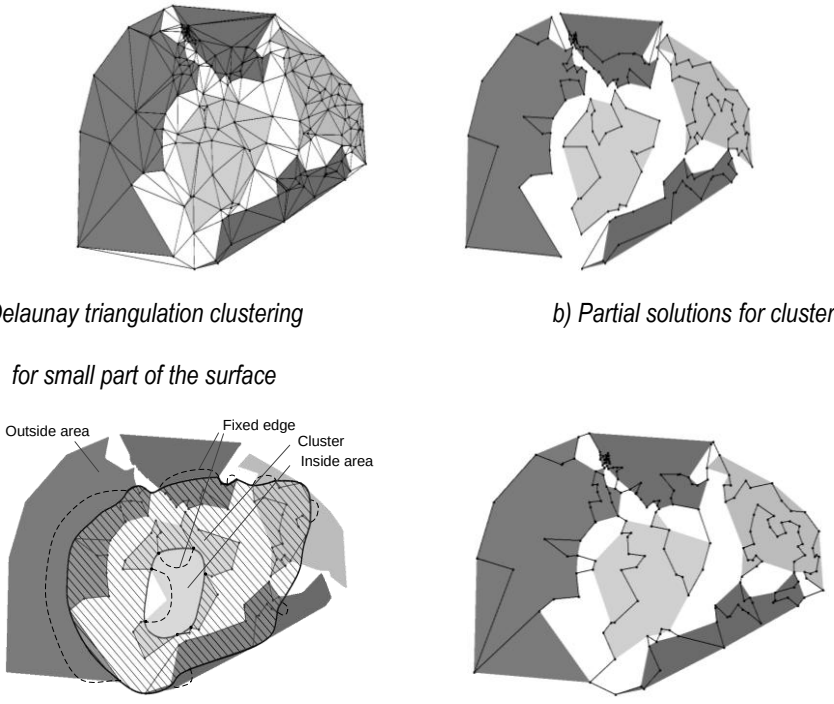
b) Choosing some neighboring area of nearly the same number of points of the previously chosen area (with given percentage of overlapping points, for example, 20% – 50 %).

c) Solving the TSP for a newly chosen area by replacing the rest pieces of already existing route with short (fixed) connections.

d) Finding the complete initial solution by sequential merging of one subset with the one adjacent to it. The extension (sequential merging) can be done in different ways – from the left to the right, from the top to the bottom, inner or outer spiral, etc.

e) Optimizing the solution.

The "Ring" method



Delaunay triangulation clustering for small part of the surface

b) Partial solutions for clusters

c) "Ring" for one cluster

d) Complete initial solution

Fig. 3. "Ring" method for TSP

The main steps of the "Ring" method [Bazylevych, 2011] developed for wide parallelization are (Figure 3, only

small part of the surface with a few clusters is considered):

a) Delaunay triangulating the set of points while considering the distances between them.

b) Clustering the set of all points using the wave propagation by triangles. Creating the clusters with given number of points.

c) Finding the partial solutions for all clusters.

d) Finding the complete initial solution by merging the partial solutions of all adjacent clusters together. For every cluster is formed a “ring” from it’s and border points of all adjacent clusters. The pieces of routes with non-considered points are replaced with the “fixed” routes.

e) Optimizing the solution.

The steps b), c) and d) can be executed in parallel.

Methods for solution optimization

Further optimization of the initial solution can be achieved by reducing the route length in the Local Optimization Areas (LOAs) [Bazylevych, 2007, 2008, 2009, 2011]. Every LOA consists of a small number of points located in a close proximity. For every LOA the solution can be obtained in a short amount of time. We developed a few optimization methods:

1. Scanning along the route (Figure 4).
2. Geometrical scanning of the whole surface (Figure 5).
3. Scanning around the cluster perimeters (Figure 6).
4. For selected “critical” areas (Figure 7).

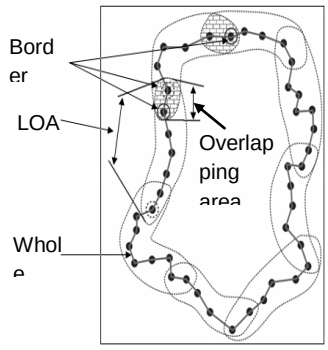


Fig. 4. Scanning along the route

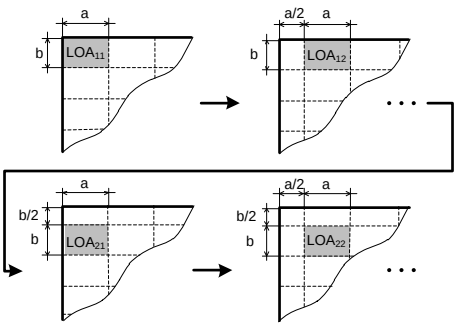


Fig. 5. Geometrical scanning of the whole surface

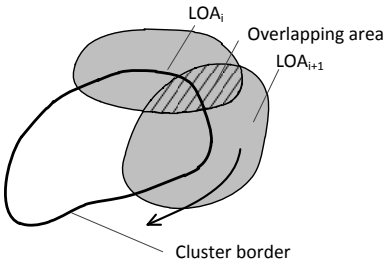


Fig. 6. Scanning around the clusters perimeters

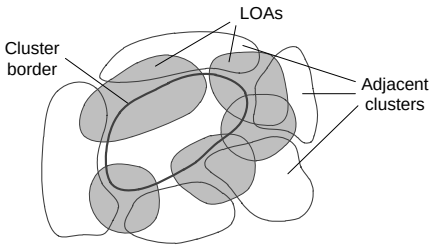


Fig. 7. Selected "critical" areas

The possible detailed changes to the route position due to optimization in LOAs are presented in the Figures 8 and 9.

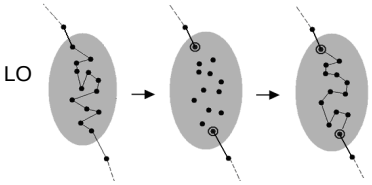


Fig. 8. Optimization in the LOA by scanning along the route

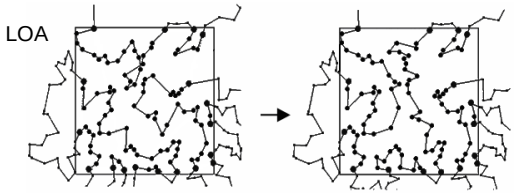


Fig. 9. Optimization in the LOA by geometrical scanning

A very important feature of the proposed optimization methods is that the adjacent LOAs have common overlapping areas. The quality of solution grows with increasing their sizes. The Scanning along the route method has two options. For the first one we consider only the rout points in the given LOA, as well as for improving the solution in the second variant - more qualitative, we also consider adjacent points with forming an enlarged zone [Bazylevych, 2009]. The quality of results depends from the LOAs and overlapping areas sizes. We studied the results depending on the sizes of overlapping areas within 20% - 80% of the LOAs sizes.

Experimental results

The proposed approach was investigated to study the solution quality and runtime. The TSP instances were taken from [TSP Art Instances]. Experiments were conducted on a PC with Athlon II X2 240 processor, 2.8 GHz CPU, and 2 GB RAM. The following parameters were used in our experiments:

- the number of points in a cluster is 800-900;

- the internal depth of a “ring” is 10 triangles (the ring covered 10 Delaunay triangles while the wave propagating inside the given cluster);
- the external depth of a “ring” is 15 triangles (the ring covered 15 Delaunay triangles while the wave propagating outside the given cluster);
- the number of points in the local optimization area is 800;
- the number of points in the overlapping area is 400.

The results are provided in Table 1. All four optimization strategies were used sequentially, i.e., one strategy after the other. According to this table, the “Tour quality %” column shows the divergence in the solution quality between the developed approach and the best known. “Time” column shows the runtime required to find the optimized complete solution. A significant feature of the developed approach is its computational complexity, which is close to linear. These results demonstrate that the developed approach can be used for large-scale problems to get high quality solutions in a reasonable amount of time. For the TSP with 200K points, the solution quality is only 0.03463% less comparatively with the best known. All experiments were executed using the LKH algorithm [Helsgaun, 2000, 2006] as the basis for all subsets.

Table 1. Experimental results

Test-case	Problem size (number of points)	Length of the initial solution	Length of the optimized solution	Time (minutes)	Length of the best known solution	Tour quality %
mona-lisa100K	100000	5 758 988	5 757 516	121	5 757 191	- 0.00565
vangogh120K	120000	6 545 620	6 544 127	178	6 543 610	- 0.00790
venus140K	140000	6 812 666	6 811 271	213	6 810 654	- 0.00905
pareja160K	160000	7 622 498	7 620 636	229	7 619 953	- 0.00896
courbet180K	180000	7 891 519	7 889 462	280	7 888 733	- 0.00924
earring200K	200000	8 174 726	8 174 507	295	8 171 677	- 0.03463

Conclusion

The large-scale intractable combinatorial NP-hard problems must have specially developed approaches to receive the high quality solutions. It was proposed to divide the problem solving into two main stages: receiving the initial solution and its optimizing. Four decomposition methods were proposed to get the initial solutions: the macromodeling for the clustered TSP, the extending partial solutions method, and the “Ring” method for the arbitrary TSP. Last of them is appropriate for wide parallelization. A few methods for solution optimization were developed: scanning along the route,

geometrical scanning of the whole surface, scanning around the clusters perimeters, and for selected "critical" areas. Suggested approaches have computation complexity close to linear what makes them suitable for large-scale problems. They provide substantial reduction in running time in comparison with the best currently known TSP heuristics. The quality loss is small (≈ 0,006% – 0,03% for 100,000 - 200,000-points instances).

Further work will be directed towards improving the solutions quality, reducing the running time by using better basic algorithms as well as towards developing some new efficient methodologies, especially for the parallel optimization algorithms.

Bibliography

[Applegate, 2009] D. Applegate, R. Bixby, V. Chvátal, W. Cook, D. Espinoza, M. Goycoolea, K Helsgaun: Certification of an Optimal TSP Tour through 85,900 Cities, Operations Research Letters, No. 37, pp. 11-15, 2009.

[Bazylevych, 2007] R. Bazylevych, R. Kutelmakh, R. Dupas, L. Bazylevych: Decomposition Algorithms for Large-Scale Clustered TSP, Proceedings of the 3rd Indian International Conference on Artificial Intelligence, Pune, India, pp. 255-267, 2007.

[Bazylevych, 2008] R. Bazylevych, R. Kutelmakh, B. Prasad, L. Bazylevych: Decomposition and Scanning Optimization algorithms for TSP, Proceedings of the International Conference on Theoretical and Mathematical Foundations of Computer Science, Orlando, USA, pp. 110-116, 2008.

[Bazylevych, 2009] R. Bazylevych, B. Prasad, R. Kutelmakh, R. Dupas, L. Bazylevych: A Decomposition Algorithm for Uniform Traveling Salesman Problem, Proceedings of the 4th Indian International Conference on Artificial Intelligence, Tumkur, India, pp. 47 59, 2009.

[Bazylevych, 2011] R. Bazylevych, R. Dupas, B. Prasad, B. Kuz, R. Kutelmakh, L. Bazylevych: A Parallel Approach for Solving a Large-Scale Traveling Salesman Problem. Proc. of the 5-th Indian Intern. Conf. on Artificial Intelligence, IICAI-2011, India, Dec., 2011, pp. 566- 579.

[Concorde] http://www.tsp.gatech.edu/concorde.html

[Delaunay, 1934] B. Delaunay: Sur la sphère vide, Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk, No 7: pp. 793–800, 1934.

[Helsgaun, 2000] K. Helsgaun: An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic, European Journal of Operational Research 126 (1), pp. 106-130, 2000.

[Helsgaun, 2006] K. Helsgaun: An Effective Implementation of k-Opt Moves for the Lin–Kernighan TSP Heuristic, Datalogiske Skrifter (Writings on Computer Science), No. 109, Roskilde University, 2006.

[Reinelt, 1994] G. Reinelt: The Traveling Salesman Problem: Computational Solutions for TSP Applications // Lecture Notes in Computer Science. – 840, Springer-Verlag. – Berlin. – 1994.

[Rohe, 1997] A. Rohe: Parallel Lower and Upper Bounds for Large TSPs // ZAMM. – 1997. – 77(2). – P. 429-432.

[TSP Art Instances] http://www.tsp.gatech.edu/data/art/index.html

[Yll Haxhimusa, 2009] Yll Haxhimusa, Walter G. Kropatsch, Zygmunt Pizlo, and Adrian Ion. Approximative Graph Pyramid Solution of the E-TSP. Image and Vision Computing, 27(7):887–896, 2009.

Authors' Information

Roman Bazylevych – *Prof. dr.hab., Mathematics and Computer Science Foundations, University of Information Technology and Management in Rzeszow;*

email:rbazylevych@wsiz.rzeszow.pl;

Major Fields of Scientific Research: Computer science, Design automation, Algorithms, Combinatorial optimization

Marek Pałasiński *– Prof. nadzw. dr.hab., Chair of Mathematics and Computer Science Foundations, University of Information Technology and Management in Rzeszow;*

e-mail: mpalasinski@wsiz.rzeszow.pl;

Major Fields of Scientific Research: Theoretical computer science, Theory of algorithms, Graph theory, Data mining and Algebraic logic

Roman Kutelmakh *– Assistant Professor, Ph.D., Software Engineering Department, Lviv Polytechnic National University, 12 S.Bandery Str., Lviv, 79013, Ukraine;*

e-mail: rkutelmakh@ua.fm;

Major Fields of Scientific Research: Software technologies, Combinatorial optimization

Bohdan Kuz *– PhD student, Software Engineering Department, Lviv Polytechnic National University, 12 S.Bandery Str., Lviv, 79013, Ukraine;*

e-mail: bohdankuz@gmail.com;

Major Fields of Scientific Research: Software technologies, Combinatorial optimization

Lubov Bazylevych *– senior scientist, Institute of Mechanical and Mathematical Applied Problems of the Ukrainian National Academy of Sciences;*

e-mail: lbaz@iapmm.lviv.ua;

Major Fields of Scientific Research: Applied mathematics and mechanics, Combinatorial optimization, Computer science

STUDY THE QUALITY OF GLOBAL NEURAL MODEL WITH REGARD TO LOCAL MODELS OF CHEMICAL COMPLEX SYSTEM

Grzegorz Drałus

***Abstract**: In the paper global modeling of complex systems with regard to quality of local models of simple plants is discussed. Complex systems consists of several sub-systems. As a global model multilayer feedforward neural networks are used. It is desirable to obtain an optimal global model, as well as optimal local models. A synthetic quality criterion as a sum of the global quality criterion and local quality criteria is defined. By optimization of the synthetic quality criterion can be obtained the global model with regard to the quality of local models of simple plants. The quality criterion of the global model contains coefficients which define the participation of the local quality criteria in the synthetic quality criterion. The investigation of influence of these coefficients on the quality of the global model of the complex static system is discussed. The investigation is examined by a complex system which is composed from two nonlinear simple plants. In this paper complex system means real chemical object (i.e. a part of the line production of ammonium nitrite).*

***Keywords**: complex system, neural network, global modeling*

***ACM Classification Keywords**: I.2.6 ARTIFICIAL INTELLIGENCE, Learning - Connectionism and neural nets*

Introduction

In the area the design of complex control systems, there are numerous difficulties associated with constructing appropriate models of complex systems and determining their parameters. One of the basic issues to be considered a model of system as a whole, i.e. to develop a global model and ensuring the quality of approximations of system components, e.g. the development of local models.

The classic task of modeling a complex system is to find optimal values of parameters of adopted mathematical model based on the established quality criteria. Mathematical methods of identification of complex objects are based on the distribution of components. The next step is to construct models of individual components (i.e. simple objects) and search for them optimal parameters. The next step is the submission of the optimal

models of simple models of the complex system [Bubnicki, 1980]. Obtained in this way model is not globally optimal, because while searching of the parameters of simple models do not take into account the interaction of components of a complex system during the modeling process. In this case we are dealing with a local modeling. The opposite approach to the local modeling is a global modeling of complex systems [Dralus and Swiatek 2000(1)], [Świątek, 2004].

Application of neural networks that have the ability to approximate nonlinear functions [Hornik, 1989] allow us to build and determine a global model parameters. The assumption of a global model to reflect the structure of a complex system in the work, and reflect the interactions of the components of a complex system during determination of model parameters. This allows us to build a more accurate model than the decomposition method [Dahleh and Venkatesh, 1997], [Dralus and Swiatek, 2000(1)], [Drałus and Świątek, 2009].

Modeling of static complex objects

Complex system are difficult to model. In principle, the methods of mathematics are not capable of modeling complex objects. To make this possible a complex system should be decomposed to simple objects [Bubnicki, 1980]. Then, separate simple objects can be modeled as independent by any methods for simple objects without considering the fact that they are part of the complex system. After obtaining the optimal parameters of simple models, assembles the complex model, which corresponded to an complex system structurally. Created in this way complex model is locally optimal, but it is not globally optimal. New fields and modern tools allow us to build global models without their decomposition. One of these tools are neural networks that allow to build a global model, which corresponds to the structure of a complex system.

By learning neural networks can to obtain satisfactory parameters of the model. Complex systems can have a varied structure. In this paper, the complex system has a cascade structure, which often occurs in industrial factories.

Global model taking into account local models

A complex system, which consists of cascaded in series *R*-th simple plants is shown in Figure 1. Simple plants are designated as O1, ..., O_R. The global model structure should correspond to the structure of the complex system. Thus, a global model has a cascade structure, and consists of *R*-th simple models designated as M_r. In a global model (see Figure 1) the *r*-th output of simple model M_r is the input to the next simple model M_{r+1}, as

in the complex system. On the other hand, in the global model, besides simple models, local models can be distinguished. Then, the output of the simple object O_r is the input to the next local model M_{r+1}.

Physically the simple model M_r and the local model M_r is the same model (one set of neural network weights). They only differ in the input signals and way of learning. Local models will be used to build a global model taking into account the quality of local models.

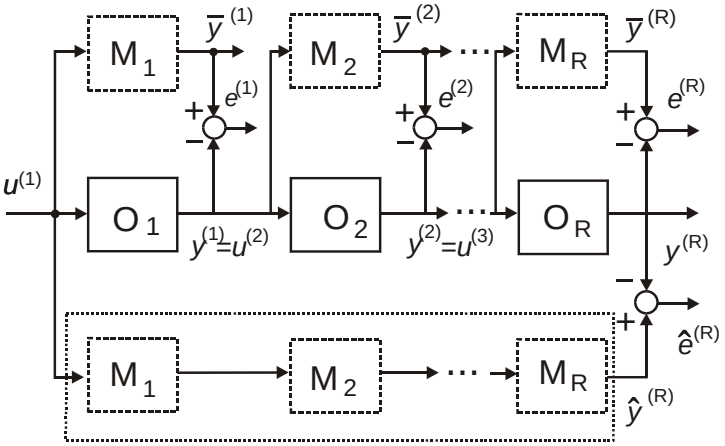


Fig. 1. Block diagram of complex system and its global model

In the global model, for each *r*-th local model is defined a local quality index as a difference between the output $\bar{\boldsymbol{y}}^{(r)}$ of the *r*-th local model and the output $\boldsymbol{y}^{(r)}$ of the *r*-th simple object:

$$Q^{(r)}(\boldsymbol{w}^{(r)}) = \frac{1}{2}\sum_{k=1}^{K}\sum_{j=1}^{J_r}\left(\bar{y}_j^{(r),k} - y_j^{(r),k}\right)^2 \tag{1}$$

where: $\boldsymbol{w}^{(r)}$ – weights of the *r*-th model simple/local model, *K* – a number of patterns J_r – a number of outputs of the *r*-th object.

For simplicity, in the global model is defined only one global quality index as a difference between the output $\hat{\boldsymbol{y}}^{(R)}$ of the *R*-th simple model and the output $\boldsymbol{y}^{(R)}$ of the *R*-th simple object:

$$Q(\boldsymbol{W}) = \frac{1}{2}\sum_{k=1}^{K}\sum_{j=1}^{J_R}\left(\hat{y}_j^{(R),k} - y_j^{(R),k}\right)^2 \tag{2}$$

where: $\boldsymbol{W}$ – weights of a global model, K – a number of patterns, J_R – a number of outputs of R-th object.

As a reminder, the output $\hat{\boldsymbol{y}}^{(R)}$ of the R-th simple model is the output of the global model, which corresponds to the output $\boldsymbol{y}^{(R)}$ in the complex system.

On the basis quality indices $Q^{(r)}$ of local models and the global quality index Q was formulated synthetic quality criterion of the global model with regard to the quality of local models. Thus, the synthetic quality criterion may take the form of a weighted sum indices of quality:

$$Q_s(\boldsymbol{W}) = \alpha_0 Q(\boldsymbol{W}) + \sum_{r=1}^{R} \alpha_r Q^{(r)}(\boldsymbol{w}^{(r)}) \tag{3}$$

where: $\boldsymbol{W}$ – weights of a global model, α_r – weighting coefficients of local models, such that $0 \le \alpha_r \le 1$, $\sum_{r=1}^{R} \alpha_r = 1$, α_0 – weighting coefficients of a global model, such that $0 \le \alpha_0 \le 1$.

The weighting coefficients α_r determine an individual participation of the quality indices $Q^{(r)}$ of local models in the synthetic quality criterion (3), while the weighting coefficient α_0 determines the participation the global quality index Q in this synthetic quality criterion Q_s. There are others method to take into account quality local models in a global model, for example a penalty function [Dralus and Swiatek, 2002].

By minimizing the synthetic quality criterion Q_s can be calculated parameters of the global model.

Backpropagation learning algorithm for a global model with regard to local models

A multilayer neural network is a global model therefore to minimize the global quality index (3) a learning algorithm is derived based on back propagation of errors. The base learning algorithm for multilayer networks is the gradient descend. According to this algorithm an increment of weight for criterion (3) is calculating as:

$$\Delta w_{ji} = -\eta \frac{\partial Q_s(\boldsymbol{W})}{\partial w_{ji}} \tag{4}$$

Calculations of the gradient (3) of the global criterion led to a new complex backpropagation learning algorithm. However, the speed of that learning algorithm according to the gradient is small and depends on the choice of learning rate η. Other algorithms are much faster. One of them is Rprop algorithm [Riedmiller and Branun, 1992]. As in the case the complex gradient algorithm**,** the algorithm Rprop has been modified and adapted for the learning of complex neural networks, having the structure of the global model called the complex Rprop [Dralus and Swiatek, 2000 (2)].

Changing the weights in the complex Rprop learning algorithm in following layers:

- in the output layer:

$$\begin{aligned}\Delta w_{ji}^{(R),M} &= -\eta_{ji}^{p}\,\mathrm{sgn}\left(\sum_{k=1}^{K} f'(z_j^{M,k})\alpha_R\left(\bar{y}_j^{(R),k} - y_j^{(R),k}\right)\hat{u}_i^{M-1,k}\right) \\ &- \eta_{ji}^{p}\,\mathrm{sgn}\left(\sum_{k=1}^{K} f'(z_j^{M,k})\alpha_0\left(\hat{y}_j^{(R),k} - y_j^{(R),k}\right)\hat{u}_i^{M-1,k}\right)\end{aligned} \tag{5}$$

- in the hidden layers:

$$\Delta w_{ji}^{(r),m} = -\eta_{ji}^{p}\,\mathrm{sgn}\left(\sum_{k=1}^{K} f'(z_j^{m,k})\sum_{l=1}^{I_{m+1}}\delta_l^{(r),m+1,k}\, w_{lj}^{(r),m+1}\,\hat{u}_i^{m-1,k}\right) \tag{6}$$

- in the "binding" hidden layers, i.e. in output layers of the simple models of a complex model:

$$\Delta w_{ji}^{(r),m} = -\eta_{ji}^{p}\left(\sum_{k=1}^{K} f'(z_j^{m,k})\left(\sum_{l=1}^{I_{m+1}}\delta_l^{(r+1),m+1,k}\, w_{lj}^{(r+1),m+1} + \alpha_r\left(\bar{y}_j^{(r),k} - y_j^{(r),k}\right)\right)\hat{u}_i^{m-1,k}\right) \tag{7}$$

In this algorithm, the learning speed ratio η is adaptive and in *p*-th step of learning is:

$$\eta_{ji}^{p} = \begin{cases} \min\left(a\cdot\eta_{ji}^{(p-1)}, \eta_{\max}^{p}\right) & \text{if} \quad S_{ji}^{p}\cdot S_{ji}^{(p-1)} > 0 \\ \max\left(b\cdot\eta_{ji}^{(p-1)}, \eta_{\min}^{p}\right) & \text{if} \quad S_{ji}^{p}\cdot S_{ji}^{(p-1)} < 0 \\ \eta_{ji}^{(p-1)} & \text{if} \quad S_{ji}^{p}\cdot S_{ji}^{(p-1)} = 0 \end{cases} \tag{8}$$

where: $S_{ji}^{p} = \dfrac{\partial Q_s(W(p))}{\partial w_{ji}}$; $\eta_{\max} = 50$; $\eta_{\min} = 10^{-6}$; a=1,2; b=0,5 [Zell, 1993].

The complex Rprop algorithm was used to learning neural networks, of which is built a global model and local models.

Simulations

For simulations was chosen a complex chemical object. This object is the production line of ammonium nitrite. It consists of several parts but, only the first two objects was selected for modeling (see Figure 2). So, the complex system for simulation consists of two simple non-linear objects connected in series.

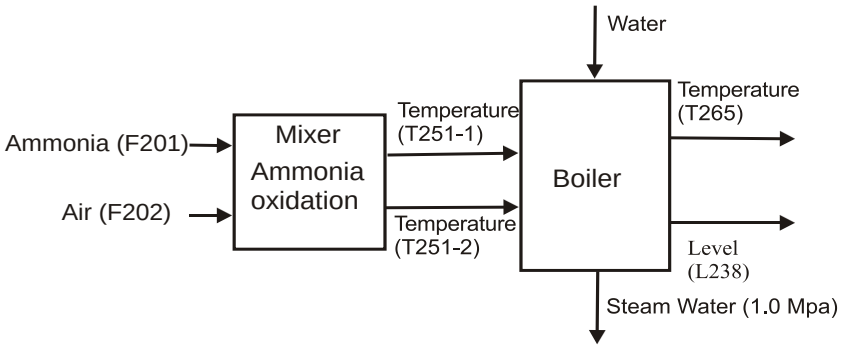


Fig. 2. The part of the installation for the production of ammonium nitrite

In Figure 3 is shown a simplified block diagram of the chemical object as a part of the production line of ammonium nitrite. In the block diagram is omitted immeasurable signals (water, steam), treated it as a constant disturbance. Input signals for the complex system are: u_1 - the flow of ammonia (F201), u_2 - the air flow (F202). The output of the first object is the input of the second object: $y_1^{(1)}$ - temperature (T251-1), $y_2^{(1)}$ - temperature (T251-2). The outputs of the second object are: $y_1^{(2)}$ - the temperature in the boiler (T265); $y_2^{(2)}$ - the solution level in the boiler (L238). These data are also the output of the complex system.

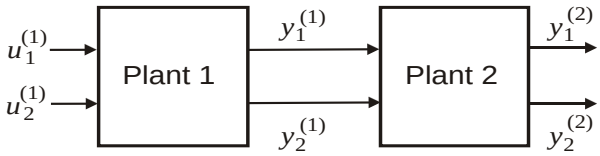


Fig. 3. Block diagram of the chemical object

Measuring learning and testing data include four days from instantaneous reports, recorded every two hours, come from the industrial production line of ammonium nitrite. The learning data was created by combining data from two days, they contain 24 items. Data from two subsequent days are the testing data. For simplified chemical object shown in Figure 3 was built a global model from a neural network, which is shown in Figure 4.

The neural model is a complex model, which corresponds to the structure of the complex system. The neural model has the following structure: 2-7T-4T-2L-7T-4T-2L. The complex model is divided into two simple models of the structure: 2-7T-4T-2L, connected in series. Simple models are simultaneously the local models and have two hidden layers with nonlinear activation functions of hyperbolic tangent (T), in the output layer the activation function is linear (L).

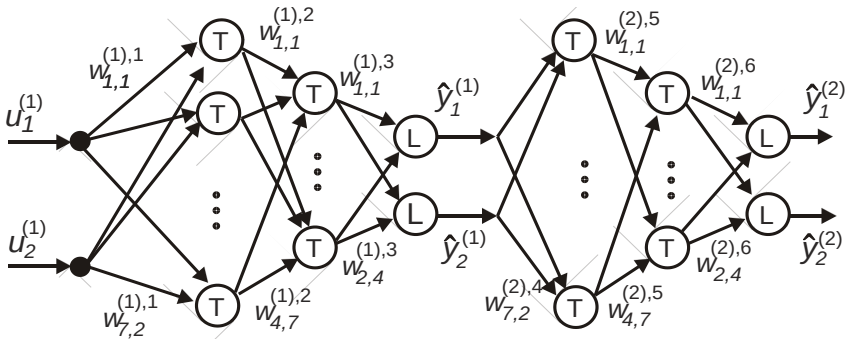


Fig. 4. The structure of the global model built from neural networks

The learning data for the neural network derived from a real object, they have a very large range, e.g. air flow hovers around 5900, and the level of the liquid solution oscillates near value of 70. Thus, all data must be scaled to the range [0..1], in which work functions of activation. Scaling was based on dividing the input and output by the maximum value appropriate for the individual data according to the formula:

$$y_j^s = \frac{y_j}{y_{\max}} \tag{9}$$

:

Where: y_j^s - is the scaled *j*-th component of the vector $\boldsymbol{y}$, y_j - the current value of the *j*-th element of the vector, $y_{\max}$ - value of the largest element in the vector $\boldsymbol{y}$.

An additional criterion for assessing the quality of modeling was adopted relative percentage error, abbreviated RPE, calculated for the *j*-th output of the *r*-th simple model with respect to the corresponding outputs a simple object:

$$\text{RPE}_j^{(r)} = \frac{\sum_{k=1}^{K} \left| \hat{y}_j^{(r)}(k) - y_j^{(r)}(k) \right|}{\sum_{k=1}^{K} \left| y_j^{(r)}(k) \right|} \cdot 100\% \tag{10}$$

Table 1. The values of quality indices for learning and testing data after 500 epochs of learning for the coefficient α_0 =1.

α_1	$Q^{(1)}$	$Q^{(2)}$	Q	Q_s	$Q^{(1)}$	$Q^{(2)}$	Q	Q_s
	For learning data				For testing data			
0,001	9.8E+6	124000	105	134000	1.8E+6	91200	112	92900
0,01	116000	350	118	1630	3430	116	114	206
0,1	2470	120	108	463	3170	122	126	490
0,2	3170	122	126	858	2070	117	108	562
0,3	2100	116	113	824	2050	118	119	757
0,4	2070	117	108	1010	2010	117	109	929
0,5	2190	117	108	1260	2210	117	109	1220
0,6	2050	118	119	1400	1960	116	113	1280
0,7	1950	116	110	1510	1950	120	119	1460
0,8	2010	117	109	1740	2000	115	112	1680
0,9	2130	116	110	2040	2030	116	114	1900
0,99	2120	117	109	2210	2010	117	116	2050
0,999	2200	117	109	2310	2020	117	116	2080

Table 2. The values of quality indices for learning and testing data after 500 epochs of learning for the coefficient α_0=0.5.

α_1	$Q^{(1)}$	$Q^{(2)}$	Q	Q_s	$Q^{(1)}$	$Q^{(2)}$	Q	Q_s
	For learning data				For testing data			
0,001	1.8E+6	91200	112	92900	1.8E+6	86000	57	87700
0,01	3430	116	114	206	4880	67	62	146
0,1	3170	122	126	490	4170	71	64	513
0,2	2070	117	108	562	1970	70	75	488
0,3	2050	118	119	757	2030	70	70	693
0,4	2010	117	109	929	1940	69	77	856
0,5	2210	117	109	1220	2310	70	69	1220
0,6	1960	116	113	1280	1920	70	71	1220
0,7	1950	120	119	1460	1850	71	71	1350
0,8	2000	115	112	1680	1970	70	59	1620
0,9	2030	116	114	1900	2010	70	69	1850
0,99	2010	117	116	2050	1980	70	73	2000
0,999	2020	117	116	2080	2000	71	71	2030

Simulations were performed for three values of the coefficient α_0 (i.e. for $\alpha_0 = 1, \alpha_0 = 0.5$ and $\alpha_0 = 0.1$) which defines the scope of the influence of the global quality index Q on the synthetic quality criterion (3). The coefficients α_r determine the influence of local quality indices on the synthetic quality criterion. Changes in the weighting factors in the synthetic quality criterion were held so that each time their sum was 1 i.e. $\alpha_1 + \alpha_2 = 1$.

The values of the quality of local models $Q^{(r)}$, the global quality index Q and the synthetic quality criterion Q_s for changes in the coefficients α_1 and α_1 for coefficient $\alpha_0 = 1$ can be found in Table 1. Simulation results for coefficient $\alpha_0 = 0.5$ are presented in Table 2 and for coefficient $\alpha_0 = 0.1$ in Table 3. Time of learning of the global model by using the complex Rprop algorithm was 500 epochs.

Table 3. The values of quality indices for learning and testing data after 500 epochs of learning for the coefficient α_0 =0.1.

α_1	$Q^{(1)}$	$Q^{(2)}$	Q	Q_s	$Q^{(1)}$	$Q^{(2)}$	Q	Q_s
	For learning data				For testing data			
0,001	116000	350	118	477	122000	306	71	435
0,01	2470	120	108	154	2270	71	71	100
0,1	2210	117	109	337	2310	70	69	301
0,2	2030	117	116	511	1990	69	71	460
0,3	1970	118	117	685	1840	70	72	608
0,4	1950	117	116	862	1880	71	72	802
0,5	1900	115	112	1020	1780	70	69	932
0,6	2100	117	116	1320	2080	70	69	1280
0,7	2000	116	114	1450	1990	70	72	1420
0,8	1970	117	115	1600	1910	70	73	1550
0,9	1930	119	118	1760	1870	70	72	1700
0,99	1960	118	118	1950	1890	70	72	1880
0,999	1960	117	116	1970	1900	70	71	1910

While the learning process of neural network the simple models interact to each other through the flow of learning signals from the input to the output, and by the flow of errors from the output layer to the input layer of the network. Local models may affect the value

of the global performance index Q . The learning process allows to determine of the global model parameters to achieve appropriate accuracy of the model. Network learning process can be terminated if the global quality index Q has achieved the desired small value. An another criterion for the termination of the learning process can be the condition that each quality index of the local model has achieved the established minimum value.

In Table 4, Table 5 and Table 6 are shown the relative percentage errors i.e. value of indices RPE for the learning data and the testing data.

Table 4. The values of RPE indices for learning and testing data after 500 epochs of learning for the coefficient α_0 =1.

α_1	RPE(1)	RPE(2)	RPE(out)	RPE(1)	RPE(2)	RPE(out)
	[%]	[%]	[%]	[%]	[%]	[%]
	For learning data			For testing data		
0,001	43.5	26.3	0.69	43.1	26.0	0.55
0,01	5.41	1.50	0.76	5.12	1.46	0.64
0,1	0.79	0.74	0.68	0.71	0.65	0.62
0,2	0.94	0.78	0.81	0.93	0.64	0.60
0,3	0.73	0.72	0.74	0.69	0.64	0.62
0,4	0.71	0.67	0.68	0.66	0.65	0.64
0,5	0.76	0.67	0.70	0.69	0.65	0.64
0,6	0.72	0.75	0.78	0.66	0.64	0.63
0,7	0.70	0.69	0.72	0.65	0.65	0.65
0,8	0.71	0.67	0.70	0.65	0.65	0.65
0,9	0.74	0.69	0.72	0.68	0.65	0.65
0,99	0.74	0.68	0.70	0.68	0.65	0.63
0,999	0.76	0.67	0.70	0.70	0.65	0.62

Table 5. The values of RPE indices for learning and testing data after 500 epochs of learning for the coefficient α_0 =0.5.

α_1	RPE(1)	RPE(2)	RPE(out)	RPE(1)	RPE(2)	RPE(out)
	[%]	[%]	[%]	[%]	[%]	[%]
	For learning data			For testing data		
0,001	25.2	25.2	0.67	25.2	24.6	0.56
0,01	0.96	0.74	0.74	1.00	0.62	0.59
0,1	0.94	0.78	0.81	0.93	0.64	0.60
0,2	0.71	0.67	0.68	0.66	0.65	0.64

Table 5 continued. The values of RPE indices for learning and testing data after 500 epochs of learning for the coefficient α_0 =0.5.

α_1	RPE(1)	RPE(2)	RPE(out)	RPE(1)	RPE(2)	RPE(out)
	[%]	[%]	[%]	[%]	[%]	[%]
0,3	0.72	0.76	0.78	0.66	0.64	0.63
0,4	0.71	0.67	0.70	0.65	0.65	0.65
0,5	0.76	0.87	0.70	0.70	0.65	0.62
0,6	0.70	0.72	0.75	0.65	0.64	0.64
0,7	0.70	0.77	0.78	0.64	0.64	0.64
0,8	0.72	0.72	0.72	0.65	0.64	0.57
0,9	0.72	0.73	0.75	0.66	0.64	0.63
0,99	0.71	0.74	0.76	0.66	0.64	0.65
0,999	0.72	0.74	0.76	0.66	0.64	0.64

The index RPE(1) expresses the quality of the first local model, the index RPE(2) expresses the quality of the second local model, and RPE(out) expresses the quality of the global model according to formula (10).

Table 6. The values of RPE indices for learning and testing data after 500 epochs of learning for the coefficient α_0 =0.1.

α_1	RPE(1)	RPE(2)	RPE(out)	RPE(1)	RPE(2)	RPE(out)
	[%]	[%]	[%]	[%]	[%]	[%]
	For learning data			For testing data		
0,001	5.41	1.50	0.76	5.16	1.46	0.63
0,01	0.79	0.74	0.68	0.71	0.65	0.63
0,1	0.76	0.67	0.70	0.70	0.65	0.62
0,2	0.72	0.74	0.76	0.66	0.64	0.64
0,3	0.71	0.75	0.77	0.64	0.63	0.65
0,4	0.70	0.74	0.76	0.64	0.64	0.64
0,5	0.69	0.72	0.74	0.63	0.64	0.62
0,6	0.74	0.75	0.77	0.67	0.64	0.63
0,7	0.72	0.73	0.75	0.65	0.64	0.64
0,8	0.71	0.74	0.76	0.65	0.64	0.65
0,9	0.70	0.76	0.78	0.64	0.64	0.64
0,99	0.70	0.76	0.78	0.65	0.64	0.64
0,999	0.71	0.75	0.76	0.65	0.64	0.64

An Influence of the coefficient α_1 on the model quality

In Table 1, Table 2 and Table 3 are shown the influence of weighting coefficients α_1 (and α_2) on the quality of local models, on the global index Q and the synthetic index Q_s for the three-values of the coefficient α_0 (for $\alpha_0 = 1$, $\alpha_0 = 0.5$ and $\alpha_0 = 0.1$) for the learning data. and the testing data. The data from Table 2 for $\alpha_0 = 0.5$ are shown in the form of graphs in Figure 5, respectively for the learning data and in Figure 6 for the testing data.

For a detailed analysis of the influence of α_1 and α_2 factors, in addition the results for the coefficient $\alpha_0 = 0.5$ are presented graphically (as the most representative).

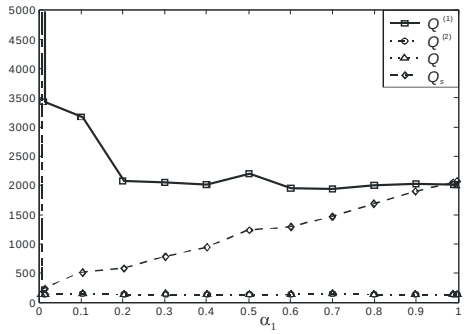


Fig. 5. Quality indices for the learning data, for coefficient $\alpha_0 = 0.5$

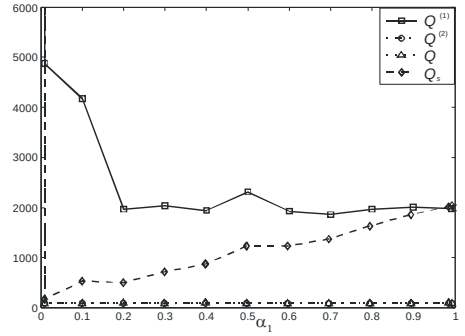


Fig. 6. Quality indices for the testing data for coefficient $\alpha_0 = 0.5$

By analyzing in detail the results e.g. for the selected value $\alpha_0 = 0.5$, can be seen that the increase in the coefficient values α_1 from 0.01 to 0.2 causes the monotonic and rapid decline in the values of quality index $Q^{(1)}$ of the first local model. Then, index $Q^{(1)}$ very slowly and a small oscillating reaches a minimum value $Q^{(1)}$=1950 for the ratio $\alpha_1 = 0.7$. For the testing data, index $Q^{(1)}$ as a function of coefficient α_1 behaves similarly. The minimum value of index $Q^{(1)}$=1850 reaches for $\alpha_1 = 0.7$ as well as for the learning data.

For the learning data, the increase in the factor α_1 (decrease α_2) the course of quality index $Q^{(2)}$ is oscillating with small fluctuations which are almost constant except for one high value for $\alpha_1 = 0.001$. The index $Q^{(2)}$ reaches a global minimum ($Q^{(2)}$=115) for

$\alpha_1 = 0.8$. For the testing data, the course of index $Q^{(2)}$ is very similar like for the learning data. The level of index $Q^{(2)}$ is somewhat lower and reaches a global minimum ($Q^{(2)}$=67) for the $\alpha_1 = 0.01$ ($\alpha_2 = 0.99$).

The global quality index Q has variable course but at the same level as the index $Q^{(2)}$. For the learning data, the index Q starts with value of $Q = 112$ for $\alpha_1 = 0.001$, and at the end of the range i.e. for $\alpha_1 = 0.999$ reaches $Q = 116$. The index Q reaches a minimum for $\alpha_1 = 0.2$, equal to Q =108. For the testing data, index Q for a change starts with the global minimum of $Q = 57$ (for $\alpha_1 = 0.001$), then its course is variable and at the end of the range of coefficient α_1 reaches $Q = 71$ ($\alpha_1 = 0.999$). For coefficient $\alpha_1 = 0.2$ reaches the maximum value of index $Q = 75$ (for the learning data at this point was the global minimum).

Global synthetic Q_s index, which is a weighted sum of $Q^{(1)}$ and $Q^{(2)}$ by α_1 and α_2, and Q by α_0 has a variable course. For the learning data, index Q_s has a high value for $\alpha_1 = 0.001$ (due to the large value of $Q^{(1)}$, see Figure 5). Starting with the $\alpha_1 = 0.01$where index Q_s has the minimum value (Q_s=206) the index Q_s increases monotonically up to a maximum value (Q_s=2080) for $\alpha_1 = 0.999$. For the testing data, the course of index Q_s is very similar to the course as for the learning data. The minimum value equal to 146 the index Q_s reaches for $\alpha_1 = 0.01$ and the maximum value equal to 2030 reaches for $\alpha_1 = 0.999$.

An Influence of coefficient α_0 on the model quality

Analysis of an influence of α_0 coefficient on a model quality is based on the results contained in all Tables.

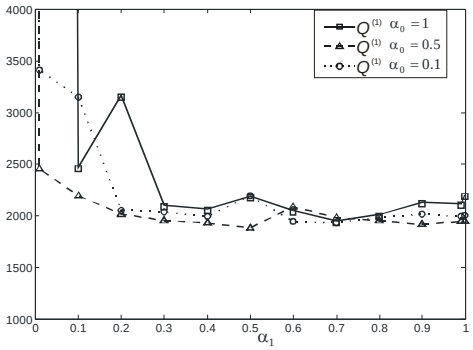


Fig. 7. Quality index $Q^{(1)}$ of first local model for the learning data

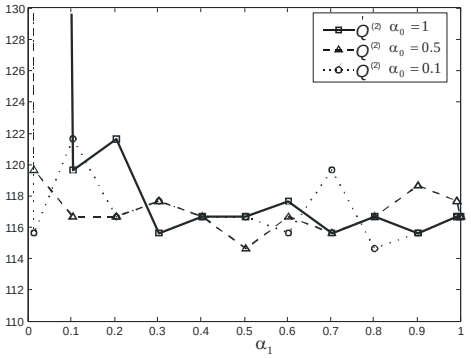


Fig. 8. Quality index $Q^{(2)}$ of second local model for the learning data

Increase of the factor α_0 that is increase participation of the global index Q in the synthetic criterion Q_s, increases the quality index $Q^{(1)}$ of the first local model except results for $\alpha_1 = 0.6$ and $\alpha_1 = 0.7$ for the learning data and testing data. So if factor α_0 increases then values of quality index $Q^{(1)}$ also increases (it is worsening, see Fig. 7).

Courses of indices $Q^{(2)}$ of second local model are oscillating and are interwoven, so it is difficult to determine the influence of the coefficient α_0 across the range α_1 variability. For example, for the selected coefficient $\alpha_1 = 0.5$ the index $Q^{(2)}$ achieves the best results for $\alpha_0 = 0.1$ which is the global minimum of index $Q^{(2)}$ for the learning data, but for the testing data the global minimum is for another value of coefficient α_1 (see Figure 8).

The global quality index Q has also variable courses. Oscillations are greatest for small values of α_1 from 0.001 to 0.3. For values of α_1 above 0.3 waveforms of index Q to stabilize and for $\alpha_1 = 0.5$ reaches a local minimum (see Figure 9). Analyzing graphs and not refer to individual deviation from the averaged values can be seen that the larger the value of factor α_0 the lower value of the index Q, for the learning data. For factor $\alpha_0 = 1$ the quality index Q has the global minimum when factor α_1 is equal 0.001 (see Figure 9)).

For the testing data, the influence of coefficient α_0 is different in different range value of α_1. For the smallest values of coefficient α_1 from $\alpha_1 = 0.001$ to $\alpha_1 = 0.3$ the lowest values the index Q takes for $\alpha_0 = 1$. For The range of values of α_1 (i.e. $\alpha_1 = 0.4$ to $\alpha_1 = 0.6$) the index Q takes the smallest value for $\alpha_0 = 0.1$.

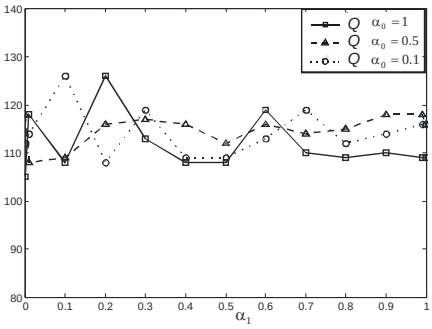


Fig. 9. Quality index Q of global model for the learning data

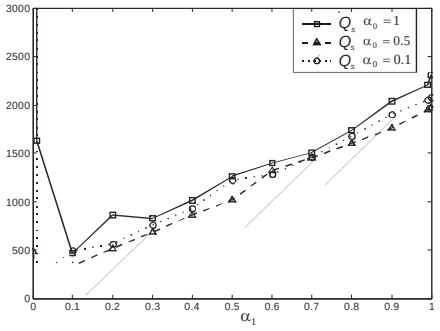


Fig.10. Synthetic quality criterion Q_s for the learning data

However, from $\alpha_1 = 0.7$ to $\alpha_1 = 0.9$ the index Q takes the lowest values for $\alpha_0 = 0.5$. At the end of coefficient $\alpha_1 = 0.99$ and $\alpha_1 = 0.999$ the index Q is the smallest for $\alpha_0 = 1$. The larger the value of coefficient α_0 that is a larger share of the global index Q in the synthetic quality criterion (3), the quality of the global model is better. For learning data, the influence of coefficient α_0 on the quality index Q can be seen more clearly and more explicitly (Figure 9). For the testing data, that influence is not as clear-cut and slightly different than for the learning data.

Synthetic index Q_s has a different course than the others. It starts from large values of $\alpha_1 = 0.001$, for coefficient $\alpha_1 = 0.01$ the index Q_s reaches the global minimum, and then increases almost monotonically with increasing α_1 until it reach the maximum for $\alpha_1 = 0.999$, depending on α_0 (see Figure 10). Changeability of the index Q_s is similar for both the learning and the testing data. The graphs clearly shows the influence of the factor α_0 on the quality index Q_s. The smaller values of coefficient α_0, the smaller the value of index Q_s for both the learning data and the testing data. This can be explained by the fact that the quality criterion Q_s is proportional to the component $\alpha_0 Q$. Thus, the higher α_0, the larger index Q_s. But we must remember Q_s

is a synthetic criterion, and depends on the values its components i.e. the indices $Q^{(1)}$, $Q^{(2)}$ and Q .

For the learning and the testing data the indicator of RPE(1) its course is almost monotone decreasing. The lowest values of the indicators RPE(1) takes for $\alpha_0 = 0.1$ and the largest takes for $\alpha_0 = 1$ except for ($\alpha_1 = 0.6$ and $\alpha_1 = 0.7$ (see Figure 11). For $\alpha_0 = 0.1$ the indicator RPE(1) reaches a global minimum for $\alpha_1 = 0.5$. At the point at which the coefficient α_1 takes the value 0.5 ($\alpha_1 = 0.5$) the indicator RPE(1) has the global minimum for all values of coefficient α_0 for the learning data and the testing data.

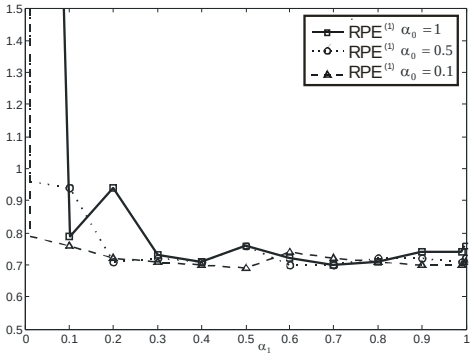


Figure 11. Quality index RPE(1) of the first local model for the learning data

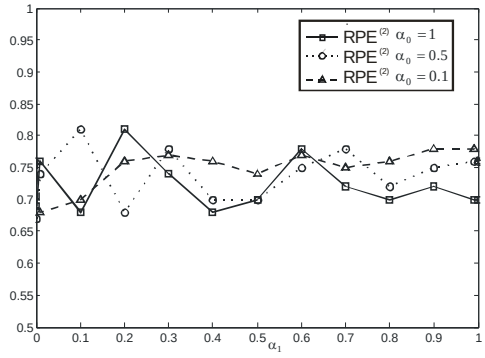


Figure 12. Quality index RPE(2) of the second local model for the learning data

The course of index RPE(2) is oscillating (see Figure 12). For the learning data, the best results i.e. lowest values of indicator RPE(2) was obtained for $\alpha_0 = 1$ and highest values of indicator RPE(2) was obtained for $\alpha_0 = 0.1$. However, for three different values of α_1 are exceptions (Figure 12). For the testing data, index RPE(2) oscillations are smaller (Table 5). However, for boundary values of coefficient $\alpha_1 = 0.001$ and $\alpha_1 = 0.999$ the best results of indicator RPE(2) are for coefficient $\alpha_0 = 1$. Generally, the best results of indicator RPE(2) was achieved for coefficient $\alpha_0 = 1$, with the exception of only two values of coefficient $\alpha_1 = 0.2$ and $\alpha_1 = 0.6$.

In Figure 13 are shown the output signals of first simple plant and simple model, and in Figure 14 are shown the output signals of second simple plant and simple model for the learning data for coefficients: $\alpha_0 = 0.5$, $\alpha_1 = 0.5$ and $\alpha_2 = 0.5$.

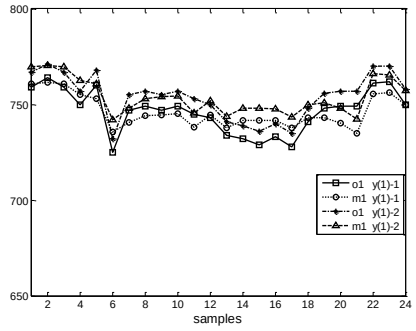


Fig. 13. The output signals of the first simple plant and simple model for the learning data

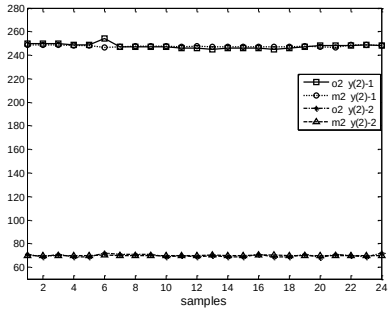


Fig. 14. The output signals of the second simple plant and simple model for the learning data

The simulations and the results show that the task of modeling complex systems is not a simple problem. The more that we had to do a simple case of a complex system consisting of two simple objects. The results obtained show relationships in the complex model, what is the quality of a global model and local models as a function of α coefficients. The complex Rprop learning algorithm, which was used to simulations also has an impact on the results, which are in some cases inconclusive. Other algorithms such as the complex Bacpropagation in conjunction with the more unequivocal complex system, which consists of two non-linear mathematical functions give a more clear results [Drałous, 2010]. However the complex Rprop algorithm in comparison with the complex Backpropagation is much faster and more reliable. Although not entirely clear results in some points, however, we can infer much about the quality of the global model taking into account the quality of local models. This knowledge can be used in other cases of modeling as well as in practice to design an optimal control of complex objects.

Conclusion

In this paper was presented the global model with respect to quality of local models of the chemical object. The global model and local models are built of multilayer neural networks. The influence of weighted coefficients α_1 and α_2 in the synthetic quality criterion (3) on the quality of the global model and the quality of local models was studied. The complex Rprop neural networks learning algorithm was used. By changing of coefficient α_0 was also studied the influence of the global quality criterion Q on

the quality of the global model and the quality of local models. The results for the learning data and the testing data was presented.

The obtained results show that by proper selection of coefficients α_1 , α_2 , and α_0 can to influence on the quality of local models and the quality of the global model. On this basis, you can specify for which values of the coefficients α_1 and α_2 , and α_0 can seek the model globally optimal.

The presented method and simulations are useful for investigation of computer control system for complex systems.

Bibliography

[Bubnicki, 1980] Z. Bubnicki. Identification of Control Plants. Elsevier, Oxford, Amsterdam, New York, 1980.

[Dahleh, Venkatesh, 1997] M.A.Dahleh, S.Venkatesh. System Identification of Complex Systems; Problem Formulation and Results, IEEE Proceedings, vol.3, pp.2441-6, 1997.

[Drałus, 2010] G.Drałus. The Investigating of Influence of Quality Criteria Coefficients on Global Complex Models. Artificial Intelligence and Soft Computing, LNAI 6114, Springer-Veralg, vol.2, pp.26-33, Berlin, 2010.

[Dralus, Swiatek, 2000(1)] G.Dralus, J.Swiatek. A modified backpropagation algorithm for modelling static complex systems using neural network. Proceedings of 5th International Conference: Neural Network and Soft Computing, pp.463-468, Zakopane, 2000.

[Dralus, Swiatek, 2000(2)] G.Dralus, J.Swiatek. Static neural network in global modeling of complex systems. Proc of Fourteen International Conference on Systems Engineering, pp.547-551, Coventry, 2000.

[Dralus, Swiatek, 2002] G.Dralus, J.Swiatek. Global network modeling of complex systems with respect of local models quality. Proceedings of Fifteenth International Conference on System Engineering ICSE'02 August 6-8 2002, pp.218-226, Univ. of Nevada, Las Vegas, 2002.

[Drałus, Świątek, 2009] G.Drałus, J.Świątek. Static and dynamic complex models: comparison and application to chemical systems, Kybernetes: The International Journal of Systems & Cybernetics, Emerald, Vol. 38, No 7/8, 2009.

[Hornik, 1989] K.Kornik. Multilayer Feedforward Networks are Universal Approximators. Neural Networks, 2, pp.359-366, 1989.

[Riedmiller, Braun, 1992] M.Riedmiller, H.Braun. RPROP – a fast adaptive learning algorithm. Technical Report, University Karlsruhe, 1992.

[Swiatek, 2004] J.Swiatek. Globally optimal model of complex systems. Proc. of Sixteen International Conference on Systems Science, vol. 1, pp.367-376, Wroclaw, 2004.

[Zell, 1993] A.Zell. SNNS – Stuttgart Neural Network Simulator, User Manual, Stuttgart, 1993.

Authors' Information

Grzegorz Drałus *– lecturer, Rzeszow University of Technology, Department of Electrical Engineering and Informatics Fundamentals, al. Powstańców Warszawy 12, 35-959 Rzeszow, Poland; e-mail: gregor@prz.edu.pl*

Major Fields of Scientific Research: Neural Networks Simulators, Complex System Modeling

ON COMBINATION OF DEDUCTION AND ANALYTICAL TRANSFORMATIONS IN E-LEARNING TESTING

Vitaly Klimenko, Alexander Lyaletski, Mykola Nikitchenko

Abstract: *We investigate a possible way for solving the problem of combination of logical inference search methods and symbolic computation tools in e-learning testing on the basis of the approaches developed at the Kiev schools of automated theorem proving and analytical transformations. The investigations started in the first half of 1960s at the Institute of Cybernetics of the Academy of Sciences of Ukraine. Some years later the Faculty of Cybernetics of the Kiev State University was involved in the corresponding projects. The current state of investigations on the topic as well as their theoretical and practical background is described in the paper.*

Keywords: *analytical transformation, automated theorem proving, deduction, e-learning, intelligent tutoring system.*

ACM Classification Keywords: *I.2.3 Deduction and Theorem Proving – Deduction. I.2.4 Knowledge Representation Formalisms and Methods – Predicate logic. G.4 Mathematical software. K.3.2 Computer and Information Science Education.*

Introduction

At the beginning of the 1960th, Academician V.M. Glushkov initiated two approaches to the development and implementation of computer-aided mathematics: one was concerned with symbolic computations (i.e. computer algebra systems in the modern terminology) and the other with automated theorem-proving (i.e. automated reasoning systems in more general sense). Now, these two approaches occupy an important place in information technologies and intelligent tutoring systems, in particular. That is why it is interesting to know what impact that their combination may have on the development of intelligent testing in e-learning in current days. In this connection, we first describe the approaches and discuss their impact after this.

Deduction in testing

Deductive testing consists in logical verification of reasoning steps expressed in a formal language. In accordance with Glushkov's paper [Glushkov, 1970], the language should be

formal and maximally be close to a natural language. Deductive testing is to be used in mathematical disciplines having the form of formal axiomatic theories containing logical inference rules. It also can be useful (within intelligent tutoring systems) for applying it in other applied domains, for example, in jurisprudence, where the testing consists in performing legally and logically valid reasoning steps, or in creating legal documents consistent with the current legislation.

Deductive approach

The deductive approach itself is based on the declarative way of representation and logical processing of knowledge having the form of formalized texts (containing axioms, definitions, propositions, and so on, when we deal with mathematical problems). Systems exploiting it usually are called automated reasoning systems or, in particular, systems for automated theorem proving. Note that this approach turns out to be the most adequate for the automated logical inference search as well as for verification of a formal text (mathematical or not), namely, checking the validity of all the reasoning steps in it.

For the purposes of deductive testing, we adhere to the following requirements for a testing environment:

- For presentation of reasoning, a trainee must use a (semi-)formal language which is close to the natural language of mathematical publications. This language preserves the structure of the problem in question and the texts in this language can be translated into some representation convenient for computer processing.
- Each reasoning step (in a natural form) from the text under verification must be "obvious" to a computer in the sense that can be checked by it. A checking procedure must evolve for incrementing reasoning steps as much as possible. It must combine general methods of logical inference search with heuristic reasoning techniques such as induction, case reasoning, definition handling, and so on. Such collection of reasoning techniques must also grow and evolve.
- Formal knowledge accumulated in the system (and used in training) must be organized in a hierarchical information environment.

The deductive paradigm is actively investigated in Ukraine from 1990, mainly at the Faculty of Cybernetics of the Kiev State University later renamed as Taras Shevchenko National University of Kyiv.

The SAD System: a Current State

As a result, the System for Automated Deduction (SAD) has been constructed. It can be downloaded or accessed online on the web-site of the Evidence Algorithm project *http://nevidal.org* (see also the papers on SAD: [Lyaletski, 2004], [Anisimov, 2006], [Lyaletski, 2010], [Lyaletski, 2006], [Vershinin, 2000]).

The SAD system conceptually consists of the following components:

- original formal language ForTheL [The Otter, 2012] that is close to the natural English language of mathematical publications; ForTheL texts can be translated into first-order language to allow automated inference search in different logics;
- special module "reasoner" that dispatches a set of traditional proving techniques of mathematical reasoning such as decomposition of a problem, simplification, reasoning by general induction, and others;
- efficient automatic provers: one of them presents the native prover Moses constructed on original sequent-based logical inference search; the other ones are the famous powerful external theorem provers such as, for example, SPASS [The SPASS, 2012], Otter [The Otter, 2012], or Vampire [The Vampire, 2012].

Note that Moses operates in natural language environment exploiting only the signature of an initial theory and, in the case of necessity, has a possibility to use tools for analytical transformations, in particular, some of the tools of the "Analitic-2007" programming system.

The SAD system was used for the formalization and verification of a number of real (non-trivial) mathematical theorems such as Ramsey's Finite and Infinite theorems, Cauchy-Bouniakowsky-Schwarz inequality, Chinese Remainder Theorem and Bezout's Identity (in terms of abstract rings), Tarski's fixed point theorem. Thus, the SAD provides a solid basis for the construction of a deductive testing system.

The following simple example presents a session of testing knowledge of a trainee in Set Theory who knows the ForTheL language. Suppose that after reading the basics of Set Theory, the trainee received the task to prove that a set being a subset of any set is empty. He has a possibility to generate the following ForTheL-text as an input text for the SAD system (containing the proposition to be proved along with its proof and all the necessary definitions, axiom, and several "explanations"):

```
Signature SetSort. A set is a notion. Let S,T denote sets.
Signature ElmSort. An element of S is a notion. Let x belongs to X stand for x is an element of X.
Definition DefSubset. A subset of S is a set T such that every element of T belongs to S.
Definition DefEmpty. S is empty iff S has no elements.
Axiom ExEmpty. There exists an empty set.
Proposition.
    S is a subset of every set iff S is empty.
Proof.
  Case S is empty. Obvious.
  Case S is a subset of every set.
    Take an empty set E.
    Let z be an element of S.
    Then z is an element of E.
    We have a contradiction.
```

Fig. 1. An example of a ForTheL-text to be verified

After checking the above-given text with the help of the SAD system using the Moses prover, the trainee will be able to know that his text is valid (that is, he correctly wrote the proof):

```
[Reason] stdin: verification successful
[Main] sections 22 - goals 6 - subgoals 10 - trivial 1 - proved 5
[Main] symbols 24 - checks 20 - trivial 20 - proved 0 - unfolds 11
[Main] parser 00:00.04 - reason 00:00.00 - prover 00:03.02/00:00.00
[Main] total 00:03.07
```

Fig. 2. The last part of the listing being generated during the SAD session

Analytical transformations in testing

This kind of testing is needed when a solution for an equation must have the form of an analytical (symbolic) expression, for example, a root of an algebraic equation or an equation in partial derivatives. In order to perform such a testing, we need a "shell" that can assure that the symbolic expression proposed by the examinee is correct, that is, it can be transformed into a formula, given by an examiner by means of symbolic computation. Such analytical verification is very appropriate for testing in various domains of physics, trigonometry, algebra, and so on. In the first place, it requires the following generic tools:

- procedures of arbitrary-precision computation for integers, rational and complex numbers;
- methods for determining whether two symbolic expressions are equal; these are usually based on various systems of term rewriting rules;

- methods of term normalization (in particular, normalization to certain conventional mathematical forms);
- tools for analytical transformations of mathematical expressions defined in terms of hierarchical data structures of arbitrary complexity.

The Institute of Problems of Mathematical Machines and Systems of National Academy of Sciences of Ukraine, (IPMMS NASU) started research in this domain in 1960s and created a family of hierarchically developing computer algebra systems in the frame of the "Analitic" project. The specialized computer series "MIR" (Mashina dlya Inzhenernyh Raschetov - Engineering Computation Machine, cf. [Glushkov, 1971],): "MIR-1", "MIR-2", and "MIR-3" having their input languages of the three first version of the "Analitic" language. Later, the developed algorithms were implemented into the SM 1410 computers ("Analitic-79" [Glushkov, 1979]) and into the standard IBM PC ("Analitic-93" [Morozov, 1995] and "Analitic-2000" [Morozov, 2001). Now, the modern project versions called "Analitic-2007" [Morozov, 2007] and "Analitic-2010" [Klimenko, 2010]. are in progress and usage. Let us give a brief description of some of their features and implementation.

Analitic-2007

The "Analitic-2007" version was implemented at the beginning of 2007 [Morozov, 2007]. It inherited all main features of all its predecessors and differed from the previous versions by deeper algebraic transformations, more detailed classification of algebraic tools, sophisticated facilities of calculations control, and improved interactive methods.

The "Analitic-2007" programming system is intended for IBM PCs and is operated as an application for the operating system Windows-98 and higher. It consists of the system kernel and a number of program packages. The compact kernel provides a user with a large quantity of programs, supports the semantic integrity of the "Analitic" language, the universality of its functional properties, and the operability of the "Analitic-2007" system in the environment of the different Windows operating systems. It performs compiling and recompiling programs and data, executing programs, transforming language objects (including programs being considered as objects of the language).

The system automatically determines the size of memory accessible for performing a program and occupies the maximal scope of accessible memory by default. A user has a possibility for determining the size of memory necessary for the normal execution of his programs. In the case of exceeding the existent memory size, the "Analitic-2007" programming system uses virtual memory.

Analitic-2010

The last version "Analitic-2010" [Klimenko, 2010] was significantly changed in its kernel and migrated on the .NET platform. A new user-friendly graphic interface missing in the previous versions was developed.

When implementing "Analitic-2010", the main efforts were directed to the improvement of operating stability of the kernel. For this the parser was recoded without any changes in the language "Analitic". As a result, the software of the previous versions was transferred to the new one.

The new interface is oriented to the efficient handling of data in the interactive mode and the faster generation of new programs. It is equipped with a complete code editor supporting intelligent input and all the possibilities inherent in a modern integrated development environment.

All the "Analitic" family systems are used actively for finding analytical solutions of tasks in mechanics, astronomy, differential equations theory. Besides, a number of experiments were performed in automated analytical transformation in various mathematical learning fields such as, for example, checking algebraic and trigonometric identities.

Combination of deduction and analytical transformations in testing

There exist a great number of software systems for authoring of "electronic textbooks" for various disciplines being taught in secondary schools, colleges, and universities. Their common feature is their orientation towards a broad spectrum of educational branches and, owing to this, towards the simplest kind of examination of trainee's knowledge based on choosing a right answer from a number of alternatives proposed by an examiner. The downside of this technique is that it gives the trainee an incentive to guess a right answer rather than really look for it, which does not allow a tutor to estimate trainee knowledge correctly. A list of "prescribed answers" is notably inconvenient for mathematical disciplines, where a solution to a problem often consists in deriving an analytic (symbolic) expression or in a formal proving when a chain of deductive steps assuring the validity of a statement under consideration must be constructed. Thus, we have the following ways for the computer-aided testing of knowledge obtained by a trainee in the (e-)learning of a subject: the query-answering method, the analytical transformation, the deductive construction, and, their combinations.

The state of the art in automated reasoning and symbolic computation has initiated transition from the simple "choose-an-answer" testing to the more intelligent and complex ones: the deductive and analytical reasoning. As it was mentioned above, the first

approach is useful in studying a mathematical theory allowing its complete formalization for computer checking logical steps of a trainee proving a certain sentence of a theory under consideration (the same concerns any knowledge domain, where formalization and deduction are admissible). The second one is suitable for testing on the base of finding analytical solutions of tasks in algebra, trigonometry, physics, and so on (for both secondary schools and higher education institutions).

We can mention a number of computer proof assistants (for example, Mizar [The Mizar, 2012]. and Isabelle [The Isabelle, 2012], some details can be found in [FmathL, 2012],) as good candidates for using deduction in testing. As to analytical transformation, there exists the great number of computer algebra systems (for example, Mathematica [Wolfram, 2003] and Reduce [20]) one of main purposes of which is to test the correctness of an analytical expression given by a trainee. But in real mathematics, a trainee is faced with texts requiring performing logical steps along with symbolic computation. This leads to the construction of tools for the combination of deduction with analytical transformations.

The last problem can be resolved by means of the "incorporation" of Analitic operators into the FortheL language for using of a linguistic extension in SAD architecture of which was designed in such a way that provided using computer algebra tools in the case of necessity [Verchinine 2007]. Of course, such a reconstruction of ForTheL and SAD will require essential efforts on the consistency of at least data formats of SAD and Analitic, but the authors are sure that moving in this direction will give a new impulse to the improvement of testing a trainee and, as a result, to the appearance on new testing standards and the increasing of e-learning quality.

Extending semantic models for the logical and analytical languages

During the last decade new approach for constructing semantic models for formal languages is being developed at the Faculty of Cybernetics of Taras Shevchenko National University of Kyiv. This approach is called a *composition-nominative approach* [Nikitchenko, 1998]. It aims to construct a hierarchy of program models of various abstraction and generality levels. The main principles of the approach are the following.

- *Development principle* (from abstract to concrete): program notions should be introduced as a process of their development that starts from abstract understanding, capturing essential program properties, and proceeds to more concrete considerations.

- Principle of *priority of semantics* over syntax: program semantic and syntactic aspects should be first studied separately, then in their integrity in which semantic aspects prevail over syntactic ones.
- *Compositionality* principle: programs can be constructed from simpler programs (functions) with the help of special operations, called compositions, which form a kernel of program semantics structures.
- *Nominativity* principle: nominative (naming) relations are basic ones in constructing data and programs.

Here we have formulated only principles relevant to the topic of the article; richer system of principles is developed in [Nikitchenko, 2009].

The above described principles specify program models as *composition-nominative systems* (CNS) [Nikitchenko, 1998]. Such a system may be considered as a triple of simpler systems: composition, description, and denotation systems. A *composition system* defines semantic aspects of programs, a *description system* defines program descriptions (syntactic aspects), and a *denotation system* specifies meanings (referents) of descriptions. Program semantics is considered as partial multi-valued functions over class of data processed by programs; compositions are *n*-ary operations over functions. Thus, composition system can be specified as two algebras: data algebra and function algebra.

Function algebra is the main semantic notion in program formalization. Terms of this algebra define syntax of programs (descriptive system), and ordinary procedure of term interpretation gives a denotation system.

CNS can be used to construct formal models of various programming, specification, and database languages [Nikitchenko, 1998], [Nikitchenko, 2009]. The program models presented by CNS are mathematically simple, but specify program semantics rather adequately; program models are highly parametric and can in a natural way represent programs of various abstraction levels; there is a possibility to introduce on a base of CNS the notion of special (abstract) computability and various axiomatic formalisms [Nikitchenko, 2001], [Nikitchenko, 2008], [Nikitchenko, 2010].

CNS are classified in accordance with levels of abstraction of their parameters: data, functions, and compositions. For constructing program models three levels of data consideration are chosen: abstract, Boolean, and nominative. At the abstract level data are treated as "black boxes", thus no information can be extracted. At the Boolean level to abstract data new data considered as "white boxes" are added. Usually, these are logical values *T* (true) and *F* (false) from the set *Bool*. At the nominative level data are considered as "grey boxes", constructed of "black" and "white boxes" with the help of naming relations. The last level is the most interesting for programming. Data of this level

are called *nominative data*. The class of nominative data is constructed inductively over a set of names V and a set of basic values W.

Concretizations of nominative data can represent various data structures, such as records, arrays, lists, relations, etc. [Nikitchenko, 1998], [Nikitchenko, 2009]. For example, a set $\{s_1, s_2, ..., s_n\}$ can be represented by a nominative data $[1 \mapsto s_1, 1 \mapsto s_2, ..., 1 \mapsto s_n]$, where 1 is treated as a standard name. Thus, the following data representation principle can be formulated: program data can be represented as concretizations of nominative data.

The above formulated levels of data abstraction may be treated as data intensions. They respectively specify three levels of semantics-based program models: abstract, Boolean, nominative. The models of each level constitute extensions of that level intension. Program models of abstract level are very poor (actually, only sequencing compositions can be defined). Program models of Boolean level are richer and permit to define structured programming constructs (sequence, selection, and repetition). This level is still too abstract and does not explicitly specify data variables. At last, models of nominative level permit to formalise compositions of traditional programming. This level (its intension) involves variables of different types.

Consider, for example, a simple educational programming language WHILE [Nielson, 2009], which is based on three main syntactic components: arithmetic expressions, Boolean expression, and statements. States of WHILE programs can be considered as partial functions from the set V of variables to the set A of basic values and here are denoted by ${}^{V}A$ $(= V \xrightarrow{p} A)$. Thus, semantics of these components is the following: arithmetic expressions specify functions of the type $Fn^{V,A} = {}^{V}A \xrightarrow{p} A$ (called partial quasiary functions), Boolean expressions define functions of the type $Pr^{V,A} = {}^{V}A \xrightarrow{p} Bool$ (partial quasiary predicates), statements specify functions of the type $Prg^{V,A} = {}^{V}A \xrightarrow{p} {}^{V}A$ (partial biquasiary functions). Note that ${}^{V}A$ is a class of single-valued nominative data. Functions over nominative data are called nominative functions. Main operations over nominative data with the name v as a parameter are naming, denaming, and checking. The main compositions (assignment, sequential, conditional, while compositions) can be formally defined over nominative functions. Obtained CNS formalize semantics of simple programming languages. Formalization of more complex languages requires more powerful classes of nominative data (hierarchic, with complex names, indirect naming, etc.) and more powerful compositions (recursive, concurrent, etc).

CNS can specify the main aspects of programming languages, and, as a consequence, it is possible to construct e-learning tools that support studying of various aspects of programming.

Based on described composition-nominative program models of various abstraction levels new logics which correspond to such models were developed. Such logics were called *composition-nominative logics* (CNL) and are oriented on program reasoning. They are logics of partial quasiary predicates and functions. Their compositions are derived from Kleene's strong connectives that permit to work with partial predicates.

Three kinds of logics can be constructed based on composition-nominative program models:

- logics, which use only partial quasiary predicates (pure predicate logic);
- logics, which use additionally partial quasiary functions (predicate-function logics);
- logics, which use also biquasiary functions (program logics).

The first type of logics generalizes classical pure predicate logics, the second type corresponds to classical predicate logic (with functions and equality), and the third type can present various logics, which use program constructs.

The following classification of these kinds of logics was proposed.

For logics of pure quasiary predicates we identify renominative, quantifier, and quantifier-equational levels.

Renominative logics are the most abstract among the above-mentioned logics. The main composition for these logics is the composition of renomination (renaming), which is a total mapping $R^{v_1,\ldots,v_n}_{x_1,\ldots,x_n} : Pr^{V,A} \xrightarrow{t} Pr^{V,A}$. Intuitively, given a quasiary predicate P and a nominative set d, the value of $R^{v_1,\ldots,v_n}_{x_1,\ldots,x_n}(P)(d)$ is evaluated in the following way: first, a new nominative set d' is constructed from d by changing the values of the names $v_1,\ldots,v_n$ in d to the values of the names $x_1,\ldots, x_n$ respectively; then predicate P is applied to d'. The obtained value of P (if it was evaluated) will be the result of $R^{v_1,\ldots,v_n}_{x_1,\ldots,x_n}(P)(d)$. For simplicity's sake the notation $R^{\bar{v}}_{\bar{x}}$ for renomination composition is also used. The basic composition operations of renominative logics are $\vee$, $\neg$, and $R^{\bar{v}}_{\bar{x}}$.

At the *quantifier* level, all basic (object) values can be used to construct different nominative sets to which quasiary predicates can be applied. This allows one to introduce the compositions of quantification $\exists x$ in style of Kleene's strong quantifiers. The basic compositions of logics of the quantifier level are $\vee$, $\neg$, $R^{\bar{v}}_{\bar{x}}$, and $\exists x$.

At the *quantifier-equational* level, new possibilities arise for equating and differentiating values using special 0-ary compositions, i.e., parametric equality predicates $=_{xy}$. Basic compositions of logics of the quantifier-equational level are $\vee$, $\neg$, $R^{\bar{v}}_{\bar{x}}$, $\exists x$, and $=_{xy}$.

All specified logics (renominative, quantifier, and quantifier-equational) are based on algebras which have only one sort: a class of quasiary predicates.

For quasiary predicate-function logics we identify function and function-equational levels.

At the *function* level, extended capabilities of formation of new functions and predicates are obtained. At this level it is possible to introduce the superposition composition $S^{\bar{x}}$ (see [Nikitchenko, 2008]), which formalizes substitution of functions into predicate (or function). It also seems natural to introduce special 0-ary compositions, called denaming functions 'x. Given a nominative set, 'x yields a value of the name x in this set. Introduction of such functions allows one to model renomination compositions with the help of superposition. The basic compositions of logics of the function level are $\vee$, $\neg$, $S^{\bar{x}}$, $\exists$x, and 'x.

At the function-equational level a special equality composition = can be introduced additionally. The basic compositions of logics of the function-equational level are $\vee$, $\neg$, $S^{\bar{x}}$, $\exists x$, *'x*, and = . At this level different classes of first-order logics can be defined.

This means that two-sorted algebras (with sets of predicates and functions as sorts and above-mentioned compositions as operations) form a semantic base for first-order CNL.

The level of *program logics* is quite rich. First, program compositions should be defined that describe the structure of programs. In the simplest case of structured programming these are:

- assignment composition AS^x: $Fn^{V,A} \xrightarrow{t} Prg^{V,A}$,
- composition of sequential execution $\bullet$: $Prg^{V,A} \times Prg^{V,A} \xrightarrow{t} Prg^{V,A}$,
- conditional composition *IF*: $Pr^{V,A} \times Prg^{V,A} \times Prg^{V,A} \xrightarrow{t} Prg^{V,A}$,
- cycling composition *WH*: $Pr^{V,A} \times Prg^{V,A} \xrightarrow{t} Prg^{V,A}$.

Let us note that above presented logics of partial predicates can be considered generalizations of classical logics. First of all, this concerns types of predicates: while classical logic is semantically based on total *n*-ary predicates, CNL are based on partial quasiary predicates, defined on a special type of nominative data. For such logics valid and complete sequent calculi were constructed [Nikitchenko, 2008]. More complex CNL are defined over hierarchic nominative data. Importance of such data is explained by their representational power that permits to model data structures of specification and programming languages. Characteristic feature of such languages is usage of composite names to access data components. The constructed logics also use composite names. On the next generalization steps modal and temporal CNL are defined and investigated [Nikitchenko, 2008].

Concerning the educational aspects of the proposed approach to formal languages specification, we can admit that this approach permits to integrate on one methodological and mathematical basis such disciplines as programming, mathematical logic, and

computability theory [Nikitchenko, 2010]. The integration is based on the idea that all these disciplines have as their kernel the notion of a specialized language system. Integration of such disciplines can be achieved by

- usage of common methodological construction principles of such disciplines;
- uniform development of main notions of disciplines;
- construction of uniform formal models of the main notions;
- constructing of the uniform e-learning tools.

This approach seems to be useful in e-learning systems because

- it is based on a small number of universal methodological principles applicable to different discipline;
- it widely uses the principle of development which proposes a number of levels starting from simple to more elaborate thus giving possibility to present more complex concepts on later stages of teaching;
- it leads to simple formal language models thus permitting their thorough investigation with further implementation of e-learning tools.

So, the constructed formal models of programming and logical languages permit to extend possibilities for deduction tools developed in Kiev by including a program reasoning component. Such extension will usually require transformation of CNL formulas into first-order classical logic [Nikitchenko, 2012].

Conclusion

The above-given analysis of Kiev approaches to symbolic computation and deduction demonstrates that the advances made by researches at IPMMS and Taras Shevchenko National University of Kyiv allow introducing and implementing various forms of distant e-learning based on a more thorough and unbiased evaluation of an examinee, which can improve the quality of learning for disciplines which admit (at least partial) formalization. Moreover, integration of analytical and deductive testing in a common framework (say, within intelligent tutoring systems) based on extended logical languages allow these two forms of intelligent testing to complement and enforce each other. Constructed tools can be incorporated into the existing e-learning systems taking into account the specifics of a domain under study. Also, one can use the proposed framework to design and implement electronic courses and textbooks, containing learning material as well as exercises for simple and intellectual testing for objective evaluation of student's knowledge.

Bibliography

[Glushkov, 1970] Glushkov V.M. Some problems of automata theory and artificial intelligence (in Russian). Kibernetika, No. 2, 1970, P. 3–13.

[Lyaletski, 2004]. Lyaletski, A., Paskevich, A., Verchinine, K.: Theorem proving and proof verification in the system SAD. In Asperti, A., Bancerek, G., Trybulec, A., eds.: Mathematical Knowledge Management: Third International Conference, MKM-04, Volume 3119 of Lecture Notes in Computer Science, Springer, 2004, P. 236–250.

[Anisimov, 2006]. Anisimov A. V. and Lyaletski A. V., (2006). The SAD system in three dimensions, Proceedings of the SYNASC'06, Timisoara, Romania, 2006, P. 85-88.

[Lyaletski, 2010]. Lyaletski A. and Verchinine K. Evidence Algorithm and System for Automated Deduction: A retrospective view. Intelligent Computer Mathematics: 10th International Conference AISC/Calculemus/MKM 2010 (Paris, France, July 2010), Vol. 6167 of Lecture Notes in Computer Science, Springer-Verlag, 2010, P. 411-426.

[Lyaletski, 2006]. Lyaletski, A., Paskevich, A., Verchinin, K.: SAD as a mathematical assistant — how should we go from here to there? Journal of Applied Logic, Vol. 4(4), 2006, P. 560–591.

[Vershinin, 2000]. Vershinin, K., Paskevich, A.: ForTheL — the language of formal theories. International Journal of Information Theories and Applications, Vol. 7(3), 2000, P. 120–126.

[The SPASS, 2012] The SPASS Prover: http://www.spass-prover.org/.

[The Otter, 2012]. The Otter automated deduction system: http://www.mcs.anl.gov/research/projects/AR/otter/.

[The Vampire, 2012], The Vampire prover: http://www.vprover.org/.

[Glushkov, 1971], Glushkov V. M., Bodnarchuk V. G., Grinchenko T. A., Dorodnizyna A. A., Klimenko V. P., Letichevsky A. A., Pogrebinsky S. B., Stogniy A. A., and Fishman Yu. S. ANALITIK (an algorithmic language for description of computational processes using analytical transformations) (in Russian), Kibernetika, No.3, 1971, P. 102-134.

[Glushkov, 1979], Glushkov V. M., Grinchenko T. A., Dorodnizyna A. A., Drakh A. M., Klimenko V. P., Pogrebinsky S. B., Savchak O. N., Fishman Yu. S., and. Tsaryuk N. P. ANALITIK-79 (in Russian), Technical report, Institute of Cybernetics, Kiev, USSR, 1979.

[Morozov, 1995]. Morozov A. A., Klimenko V. P., Fishman Yu. S., Bublik B. A., Gorovoy V. D., and Kalina E. A. ANALITIK-93 (in Russian), Kibernetika i sistemnyj analiz, No.5, 1995, P. 127-157.

[Morozov, 2001]. Morozov A. A., Klimenko V. P., Fishman Yu. S., Lyakhov A. L., Kondrashov S.V., and Shvalyuk T. N. ANALITIK-2000 (in Russian), Matematicheskie mashiny i sistemy, No. 1-2, 2001, P. 66-99.

[Morozov, 2007]. Morozov A. A., Klimenko V. P., Fishman Yu. S., and Shvalyuk T. N. ANALITIK-2007 (in Russian), Mathematical Machines and Systems, No. 3-4, 2007, P. 8-52.

[Klimenko, 2010]. Klimenko V. P., Lyakhov A.L., Gvozdik D.N., Zakharov S.A., and Shvalyuk T. N. On the implementation of a new version of the Analitic family CAS (in Russian), Proceedings of the International conference CMSEE-2010, Poltava, 2010.

[The Mizar, 2012], The Mizar system: http://www.mizar.org/.

[The Isabelle, 2012], The Isabelle system: http://www.cl.cam.ac.uk/research/hvg/Isabelle/.

[FmathL, 2012], FMathL - Formal Mathematical Language: http://solon.cma.univie.ac.at/FMathL.html

[Wolfram, 2003], Wolfram S., (2003). The Mathematica Book, Fifth Edition, Wolfram Media, Inc.

[Reduce, 2012], The computer algebra system Reduce: http://reduce-algebra.sourceforge.net/.

[Verchinine 2007], Verchinine, K., Lyaletski, A., and Paskevich, A. System for Automated Deduction (SAD): a tool for proof verification. In Automated Deduction, 21st International

Conference, CADE-21 (Bremen, Germany, July 2007), F. Pfenning, Ed., vol. 4603 of Lecture Notes in Computer Science, Springer-Verlag, P. 398-403.

[Nikitchenko, 1998], Nikitchenko, N.S.: A Composition Nominative Approach to Program Semantics. Technical Report IT–TR 1998-020, Technical University of Denmark, 103 p., 1998.

[Nikitchenko, 2009], Nikitchenko M.S., Composition-nominative aspects of address programming, Kibernetika I Sistemnyi Analiz, 2009, 6, P. 24-35 (In Russian)

[Nielson, 2009], Nielson H.R., Nielson F.: Semantics with Applications: A Formal Introduction. John Wiley & Sons Inc, 1992.

[Nikitchenko, 2001]. Nikitchenko N.S., Abstract computability of non-deterministic programs over various data structures, In: Bjørner D., Broy M., Zamulin A.V. (Eds.), Perspectives of System Informatics, LNCS, 2001, 2244, P. 471-484.

[Nikitchenko, 2008]. Nikitchenko M.S., Shkilnyak S.S., Mathematical logic and theory of algorithms, Publishing house of Taras Shevchenko National University of Kyiv, Kyiv, 2008, 528 p. (in Ukrainian)

[Nikitchenko, 2010]. Nikitchenko M.S. Integrating programming-related disciplines: main principles and notions. In: Proc. of 8th Int. Conference on Emerging eLearning Technologies and Applications. The High Tatras, Slovakia, October 28-29, 2010, P. 49–56.

[Nikitchenko, 2012]. Nikitchenko M.S., Tymofieiev V.G.: Satisfiability Problem in Composition-Nominative Logics of Quantifier-Equational Level. In: Proc. 8-th Int. Conf. ICTERI 2012, Kherson, Ukraine, June 6-10, 2012. CEUR-WS.org/Vol-848, P. 56-70.

Authors' Information

Vitaly Klimenko *– Deputy Director of the Institute of Problems of Mathematical Machines and Systems of NAS of Ukraine, 42, Acad. Glushkova Ave., 03680, Kyiv, Ukraine;*

e-mail: klimenko@immsp.kiev.ua

Major Fields of Scientific Research: Computer algebra, Mathematical software, Architecture of computer systems

Alexander Lyaletski *– Senior researcher of the Faculty of Cybernetics at the Taras Shevchenko National University of Kyiv, 64, Volodymyrska Street, 01601 Kyiv, Ukraine;*

e-mail: lav@unicyb.kiev.ua

Major Fields of Scientific Research: Automated Reasoning, Proof theory, Mathematical and Applied Logics

Mykola Nikitchenko *– Chairman of the department of theory and technology of programming at the Taras Shevchenko National University of Kyiv, 64, Volodymyrska Street, 01601 Kyiv, Ukraine; e-mail: nikitchenko@unicyb.kiev.ua*

Major Fields of Scientific Research: Foundations of informatics, Formal software system development, Mathematical logic, Computability theory, Courseware for informatics

BUSINESS INTELLIGENCE SYSTEMS

J. FORRESTER'S MODEL OF WORLD DYNAMICS AND ITS DEVELOPMENT (REVIEW)

Olga Proncheva, Sergey Makhov

Abstract: *At far 1970 the elite Roman Club asked prof. J. Forrester from MIT to develop a model of world dynamics. Speaking world dynamics we mean the dynamic interactivity of the main macro economical variables. The 1-st version of the model named "World-1" was presented in 4 weeks and next year the corrected version "World-2" was accepted as the classical J. Forrester's model. In spite of its long history the J. Forrester model retains its actuality being the basis for modern models. In the paper we consider the principal of system dynamic, criticism of the classical model, and the new models developed by the J. Forrester's followers. We consider also adjacent areas and open problems related with world dynamics.*

Keywords*: world dynamics, non lineal dynamics, J. Forrester model*

ACM Classification Keywords*: I.2.m Miscellaneous*

Introduction

1.1. The beginning of J. Forrester's model

The elite Roman Club is non-governmental organization, which joins political und scientific personalities and is working on modeling World Crisis. At 1970 the Roman Club asked prof. J. Forrester from the Massachusetts Institute of Technology (MIT) to develop a model of world dynamics. Speaking world dynamics we mean the dynamic interactivity of the main macro-economical variables. The first version of the model named "World-1" was presented in 4 weeks and next year the corrected version "World-2" was accepted as the classical J. Forrester model. In spite of its long history the J. Forrester model retains its actuality being the basis for modern models, so we can say, that this model is actual

nowadays in spite of its elderly age. The models based on J. Forrester's approach can predict crises and sometimes help to avoid it. So, such models are very important.

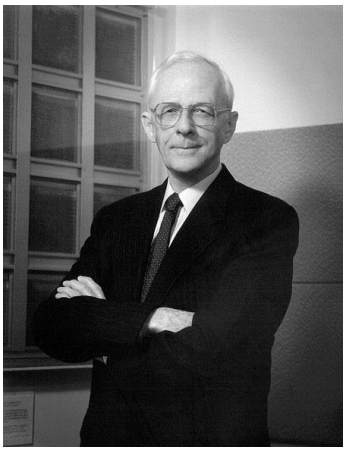


Fig.1. Prof. J. Forrester

There are many reviews in literature associated with these models [Makhov, 2003; Makhov,2005; Malineckiy, 2010, etc.]. We tried to collect shortly the principal positions reflected in the mentioned publications.

1.2. Principles of system dynamics

System dynamics is based on two main principles. First of all, equations of the same type are prepared for all variables:

$$\frac{dy}{dt} = y^{+} - y^{-}. \quad (1)$$

Here y^{+} is the positive rate of variable change (it includes all factors related with increase of the variable y), y^{-} is the negative rate of variable change (it includes all factors related with decrease of the variable y).

Thereafter it is supposed that all rates (the positive and negative ones) could be presented in the form of function compositions, which depend on one factor (combination of main variables):

$$y^{\pm} = g(y_1, y_2, \ldots, y_n) = f(F_1, F_2, \ldots, F_k) = f_1(F_1) f_2(F_2) \ldots f_k(F_k). \quad (2)$$

1.3. J. Forrester's model

J. Forrester in his work saw five main problems, which could provoke the World Crises. It is overpopulation of our planet, lack of basis resources, critical level of pollution, food shortages and industrialization and the related industrial growth. He tied a single variable

with each of these issues. So, we have a five-level system, which define the structure of the system:

• Population (P).

• Pollution (Z).

• Natural resources (R).

• Fixed capital (K).

• Capital investment in agriculture fraction (X).

For the system level J. Forrester proposed the following differential equation:

$$\frac{dP}{dt} = P(c_B B_C B_P B_F B_Z - c_D D_C D_P D_F D_Z) \quad (3)$$

$$\frac{dK}{dt} = c_K P K_C - \frac{K}{T_K} \quad (4)$$

$$\frac{dX}{dt} = \frac{X_F X_Q - X}{T_X} \quad (5)$$

$$\frac{dZ}{dt} = P Z_K - \frac{Z}{T_Z} \quad (6)$$

$$\frac{dR}{dt} = -P R_C \quad (7)$$

Here he used tabulated functions (with linear interpolation) and constants: c_B=0,04 (normal fertility rate), c_D=0,028 (normal death rate), c_K=0,05 (normal rate of capital), T_K=40 (time of depreciation main funds), T_X=15 (time of depreciation agricultural funds), t_N=1970 (initial year), P_N=3,6· 10^9(population in initial year), X_N=0,3 (capital investment ratio in agriculture in initial year).

Initial data are: $t_0 = 1900$, $P_0 = 1{,}65 \cdot 10^9$, $K_0 = 0{,}4 \cdot 10^9$, $X_0 = 0{,}2$, $Z_0 = 0{,}2 \cdot 10^9$, $R_0 = 900 \cdot 10^9$.

Standard pollution Z_N is numerically equal to the population. R_0 is a rate of mineral resources consumption. It is taken under the two conditions: a) it is equal the rate in 1970 b) this rate is enough for natural resources would be sufficient for 250 years

The behavior of the model parameters is shown in figure1. It can be seen that after a period of growth, the population ***P*** begins to decline since 2020. Non-renewable natural resources in 2100 are less than 30% of the original stock. Pollution reaches its maximum in 2050, about 6 (more precisely, 5.8) times exceeding the standard level, then it drops due to the general decline of industry and population decline. Level of life reaches its maximum about 2000, and then decreases.

Such a behavior is a consequence of resource depletion. Less natural resources less the level of life. The latter causes increasing death rate and reduces investments. And, finally, we have a sharp population decline and a fall of industrial production (funds). J. Forrester tried to change the original settings in order to avoid the crisis, but every time the crisis arose.

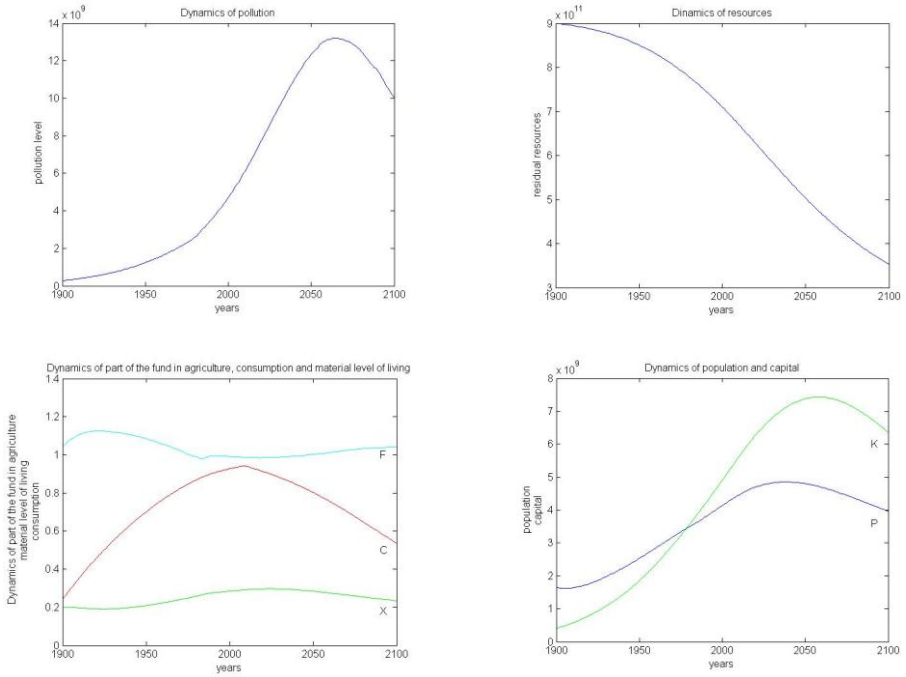


Fig.2. The behavior of the main macro economical variables

Critical analysis of J. Forrester model

Like any model, the global dynamics of J. Forrester has some limitations [Forrester, 2003]. Many researchers have noted it in their publications.

2.1. Critics by Moiseev

Moiseev's notes:

1. Methodological flaw: there are no conservation laws of material flow in the model that would reflect the economic balance
2. STP is not considered although it changes the nature of production and increasing productivity
3. There are no social mechanisms for the distribution of wealth in the model

4. Unreasonable nature of the scalar concept of a "quality of life".
5. The model is too rigid: once started "correct" model we then obtain a model of a completely different nature
6. Some table functions are incorrect (in the sense of the nature of relationships): factors may obtain other values
7. Mechanistic understanding of the concept of equilibrium, which J. Forrester recommends to prevent the crisis.

2.2. Critics by Egorov

Egorov [Egorov, 1980] and Gelovani [Gelovani, 1977] note:

1. Disaggregated model across regions is needed.
2. There is no possibility of conscious human impact on the process of development
3. Lack of any control due to the rigidity of the investment distribution
4. It is incorrect to give recommendations how to prevent a serious crisis without analysis of all possible actions
5. "Stability" offered by J. Forrester is unreachable in the framework of his rigid model

Many researchers have criticized the Forrester's model of world dynamics, mainly due to its rigidity and lack of technological factors.

Models of J. Forrester's followers

3.1. Meadows' model

One of the works, which can be considered as a development of J. Forrester's model is the work of Meadows [Meadows, 2007]. He is a pupil of J. Forrester. Meadows expanded the model having introduced several system levels in addition to those of J. Forrester. The integration of processes was completed at the same time interval. Modeling showed that on the qualitative level the new model was similar to the of J. Forrester model. But, unfortunately, Meadows had no enough quantity of data (according to him, he had only 0.1% of the required information). The results agree with Forrester's model because, restoring the missing data, Meadows was guided by the model of his teacher. By the way the Meadows' model also showed that a crisis was inevitable due to resource depletion.

3.2. Egorov's model

Modeling world dynamics allowed to formulate the simple conclusion: if the current trends of social development is inevitable then it will be a serious crisis in relations between humans and the environment. Growth can not continue indefinitely, sooner or later it will stop. But how will this growth stop? The answer of the model did not satisfy J. Forrester

and he introduced the concept of global equilibrium. However it proved to be impossible to achieve a steady state for all variables in the framework of his model. The group of researchers headed by Egorov from the Keldysh Institute of Applied Mathematics proposed a way to achieve such a balance (a stationary mode of model functioning). The main positions [Egorov, 1980] are the following:

- it is possible to recover (recycle) resources
- there are ways to reduce pollution
- there is a control of capital distribution between agriculture, resource recovery and struggle with pollution
- model is modified during its functioning
- an optimal control problem is formulated and resolved for the "corrected" model with the appropriated criterion of model quality

Speaking on the "language of equations" these positions mean the following:

a. Right part of the equation for resources is added by an additional summand:

$$\frac{dR}{dt} = -PR_C + \frac{K}{Cr}U_R \tag{8}$$

Here U_R is the part of funds, which is used for recovering resources, and C_R is the cost of repairing the unit of resource

b. The right part of the equation for pollution is added by an additional summand:

$$\frac{dZ}{dt} = PZ_K - \frac{Z}{T_Z} - \frac{K}{C_Z}U_Z \tag{9}$$

Here U_Z is the part of funds to be directed for the struggle with pollution and C_Z is the cost of cleaning the unit of pollution.

c. The right part of the equation for agricultural is added by the factor (1+U_X):

$$\frac{dX}{dt} = \frac{(1+U_X)X_F X_Q - X}{T_X} \tag{10}$$

d. The level of life is calculated by the new formula:

$$C = K_p \frac{1 - X - U_R - U_Z}{1 - X_0} \tag{11}$$

Egorov has shown that the new model has non-zero steady-state solutions. Therefore with the controlled model the world system can avoid a crisis in the sense of J. Forrester.

3.3. Matrosov's model

Another approach is proposed by a group of Matrosov [Matrosov, 1999; Matrosova 1999]. Here are the principle positions related with their model:

- the dynamics of biomass of vegetation is introduced (the model contains a "control parameter": coefficient of pollution influence on biomass)
- the single-product macroeconomic models are introduced (there are an explicit GDP), which form an expanded sector of the economy
- the time-dependent STP is introduced as an index of average productivity of labor
- the factor of political tensions and management is introduced
- we introduce the appropriate factors and parameters for modified equations

The stationary solutions for the model were calculated and they proved their stability.

Thus, the authors [Matrosov, 1999; Matrosova 1999] completed a large-scale modification of J. Forrester model. In essence, it was a new model. However, serious doubts concerning the adequacy of this model appeared. Forrester's model is "self-sufficient": the factors and parameters in it were chosen to simulate the dynamics of the past. The Matrosov's model [Matrosov, 1999; Matrosova 1999] has no this property. On the other hand, the model has a number of advantages. One of them is the detailed economical sector.

3.4. Makhov's model

The last modification of the model was proposed by one of the authors [Makhov, 2010]. His model contains the following parameters: population ***N***, energy reserves ***R***, fixed assets, similar to J. Forrester's model. But, unlike the mentioned models the new model does not include agriculture and pollution. They are replaced by the level of technology ***T*** and the education ***E***. The agriculture sector depends on the energy and capital that is it is not an independent factor. For the same reason pollution is not considered separately: it depends on territory and resources. Besides, the affect of pollution on the environment was not studied well.

The demographic equation is based on the well-known model [Makhov, 2010]:

$$\frac{dN}{dt} = c_N N\left(1 - \frac{M}{M_{max}}\right) \tag{12}$$

The equation for capital can be written as (it is a result of logical reasoning):

$$\frac{dK}{dt} = I - \mu K \tag{13}$$

Here ***I*** is the cross (fixed) capital formation, ***μ*** is the retirement rate.

The analysis of data over the past 40 years shows that the fractional amount of final consumption and cross (fixed) capital formation is only slightly deviated from 1 (see Fig.3).

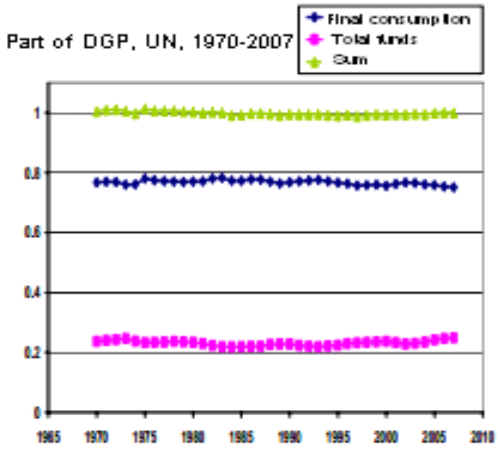


Fig.3. Parts of final consumption and total funds in GDP, and its sum

The equation for resources looks like:

$$\frac{dR}{dt} = -R_D + R_P \tag{14}$$

Here R_D is production and R_P is replenishment of resources. Energy-related part is assumed to be directly proportional to the GDP (see Fig.4).

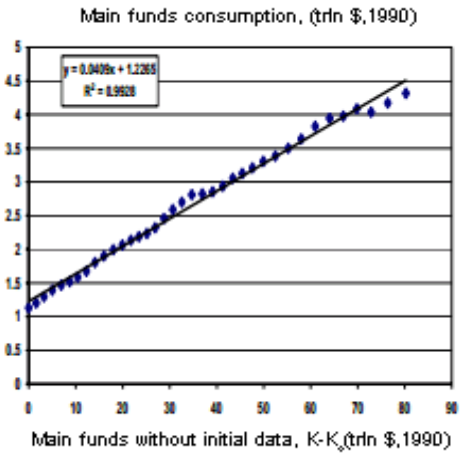


Fig.4. The main funds consumption as a function of the total gross savings in capital (the integral of gross) for 1970-2006

The equations for the sectors of education and technology are built empirically. A dependence of the population on the capital and technology is given by the Cobb-Douglas function.

Thus, the Makhov's model of world dynamics [Makhov, 2010] is presented in the form:

$$\frac{dN}{dt} = c_N N(1 - \frac{g_c Y}{N M_{max}}) \quad (15)$$

$$\frac{dK}{dt} = (1 - g_c)Y - \mu K \quad (16)$$

$$\frac{dR}{dt} = -k_R Y + R_P \quad (17)$$

$$\frac{dT}{dt} = \alpha(c_T(g_T T)^m E^n - T) \quad (18)$$

$$\frac{dE}{dt} = \beta(a + bT - E) \quad (19)$$

$$k_R = k_0 - k_1(t - t_0) \quad (20)$$

$$Y = TK^{\alpha}N^{1-\alpha} \quad (21)$$

Conclusion

In the paper we presented the review of Forrester's model and the models of his followers.

- 40 years passed but researchers continue to study and develop various models of world dynamics. Such a fact has a simple explanation: crises come now and they will come in future
- In order to build any own model of world dynamics one should study first of all the classical J. Forrester model

Bibliography

[Egorov, 1980] Egorov, V., Kallistov N., Mitrofanov, N., Piontkovsky, V. *Mathematical model of global development: a critical analysis of models of nature*. - Gidrometeoizdat, 1980. - 192. (rus).

[Forrester, 2003] Forrester J. *World Dynamics*. - Moscow: AST, 2003. - 379 p. (rus)

[Makhov, 2005] Makhov S. *Mathematical modeling of the world dynamics and sustainable development on the model of Forrester* / / Preprint. MV Keldysh RAS. - 2005. - № 6 to -24 (rus).

[Makhov, 2010] Makhov S. *Long-term trends and projections in terms of a new model of world dynamics / forecast and modeling the dynamics of the global crisis*/ Ed. A. Akayev, A. Korotayev, G. Malinetskii / Future Russia. - M.: LCI, 2010. - P. 262 - 276. (rus)

[Matrosov, 1999] Matrosov V. Matrosov I. *Global modeling taking into account the dynamics of biomass and scenarios for sustainable development / new paradigm of development in*

Russia (Comprehensive studies on sustainable development). - Moscow: Academia, MGUKI, 1999. - S. 18 – 24 (rus).

[Matrosova, 1999] Matrosova K. *Sustainable development in a modified mathematical model of the "World Dynamics" / new paradigm of development in Russia (Comprehensive studies on sustainable development)*. - Moscow: Academia, MGUKI, 1999. - S. 344-353 (rus).

[Meadows, 2007] Meadows D., Randers J. *The Limits to Growth. 30 years later*. -M.: ICC "Akademkniga", 2007. - 342 p (rus).

[Petrov, 2006] Petrov I., Lobanov A. "Lectures in Computational Mathematics." - M: The Internet University of Information Technology; BINOM. Laboratory of Knowledge, 2006. - 523 p. (rus)

[Vasiliev, 2012] Vasilev A. "Matlab. Tutorial. A Practical Approach". - Moscow: Science and Technology, 2012. - 448 p. (rus)

Authors' Information

Olga Proncheva *– M.Sc. student, Russian Presidential Academy of national economy and public administration; Prosp. Vernadskogo 82, bld. 1, Moscow, 119571, Russia; Moscow Institute of Physics and Technology (State University); Institutskii per 9., Dolgoprudny, MoscowRegion, 141700, Russia*

e-mail: olga.proncheva@gmail.com

Major Fields of Scientific Research: macroeconomics, mathematical modelling

Sergey Makhov *– Asoc. Prof., PhD, Keldysh Institute of Applied Mathematics, Miusskaya sq., 4, Moscow, 125047, Russia;*

e-mail: s_makhov@mail.ru

Major Fields of Scientific Research: mathematical modelling, sinergetics, data mining.

TESTING STABILITY OF THE CLASSICAL FORRESTER MODEL TO INITIAL DATA AND ADDITIVE NOISE

Olga Proncheva, Mikhail Alexandrov, Sergey Makhov

Abstract: *The classical Forrester model of world dynamics is a system of 5 differential equations related with 5 macro-economical variables (population, resources, etc.). This model was developed at 1970-1971 but by the moment its stability to noise was not studied. The plan of experiments is described and the results of modeling are presented. It proved that a) noise affects stronger initial data then the model during its functionality b) change of resources is the most critical value in comparison with the other system variables. All experiments have been made by means of the program WorldDyn developed on MatLab.*

Keywords*:* Forrester model, word dynamics noise immunity, numerical analysis

Introduction

At 1970 year the Roman Club (nongovernmental organization of politicians and scientists) asked John Forrester (professor of MTI) to create a model, which could predict the development of our world. In 2 weeks he presented his model “World-1” but this model proved to be too crude. At 1971 Forrester presented his second model called “World-2” and just this model is considered in the paper.

In his work Forrester selected five main problems, which could provoke World Crisis in the future. It is an overpopulation of our planet, a lack of basis resources, a critical level of pollution, food shortages and industrialization. Each of these problems was reflected in the corresponding variable:

• Population (P)

• Pollution (Z)

• Natural resources (R)

• Capital investment (fixed assets) (K)

• Proportion of funds invested in agriculture (X)

All variables were united in one system of differential equations. Forrester developed the principles of system dynamics and these principles defined the structure of the mentioned system of equations.

Forrester has many followers: Vladimir Egorov [Egorov, 1980], Sergey Makhov [Machov,2003], David Meadows [Meadows, 2007], et al. But we do not know papers where noise immunity of the model was studied. We associate the noise with inaccuracy of initial data and tuned model parameters. In the paper we study this problem under the essential restriction: the noise is considered as the additive one.

The paper consists of 3 chapters. The second chapter deals with planning experiments, the third chapter describe the experiments, and finally we give a short description of the software we used for experiment implementation.

Planning experiments

2.1 Noised model and methods of its calculation

The noised Forrester model looks like the following system:

$$\frac{dP}{dt} = P(c_B B_C B_P B_F B_Z - c_D D_C D_P D_F D_Z) + \vartheta \quad (1)$$

$$\frac{dK}{dt} = c_K P K_C - \frac{K}{T_K} + \theta \quad (2)$$

$$\frac{dX}{dt} = \frac{X_F X_Q - X}{T_X} + \mu \quad (3)$$

$$\frac{dZ}{dt} = P Z_K - \frac{Z}{T_Z} + \sigma \quad (4)$$

$$\frac{dR}{dt} = -P R_C + \tau \quad (5)$$

Here: ϑ, θ, μ, σ, τ are stationary white noise.

To calculate the model we used two subprograms from MatLab package. These programs realize the well-known Runge-Kutta method and Adams method [Petrov, 2006]. Many variables in the model, such as B_C, were presented in a tabulated form. To use them we applied the linear interpolation. The same way was used by Forrester.

Before the experiments we tested the influence of time step on the results of calculation. It proved that such an influence was inessential. For this reason we used one-year step.

2.2. Analysis of Forrester model stability

The middle value of each noise realization was equal 0. The noise dispersion ε^2 was calculated by the following way:

a) Forecast period

The dispersion is equal $\varepsilon^2 = \alpha \left[\frac{1}{71}\int_{1900}^{1970} f^2(t)dt\right]$. Here: α is a coefficient of proportionality, the value in parenthesis is a middle value of a power for the correspondent variable. The dispersion did not depend on the variable itself. Naturally, the absolute values of noise were different for different variables.

b) Initial data

The dispersion is equal $\varepsilon^2=\alpha f$. Here: α is a coefficient of proportionality, f is a value of the correspondent variable

We calculated the model 100 times for various noise realizations and fixed the number of cases with convergent processes.

2.3. Software development

To make the experiments we developed the program WorldDyn. This program has 3 options for study the noise influence on the Forrester model: noise affects initial data, noise affects all variables simultaneously on the stage of forecast, and noise affects each variable separately. The program has a convenient graphic interface. It allows to see: initial (un noised) function, all realizations of this function with a noise, and the worst realization (it has the maximal root-mean-square deviation). A help system is a part of interface. Figure 1 presents some elements of the program interface

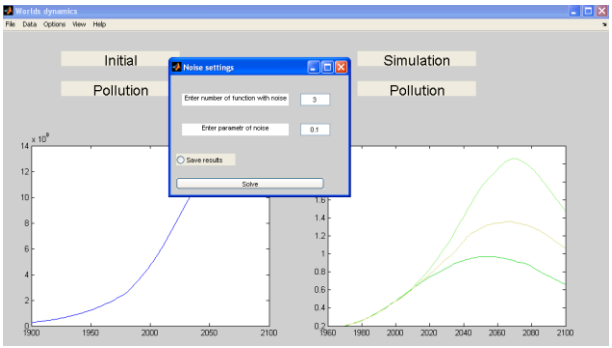


Fig. 1. Elements of WorldDyn interface

Experiments

3.1. Noise affects all variables on the stage of forecast

As an example we consider the case with the 20% noise. Figure 2 presents the results of modeling for all macro-economical variables. There are 3 lines on the figure: thin uninterrupted line is the initial function, thick line is the forecast, and thin dotted line is the worst function.

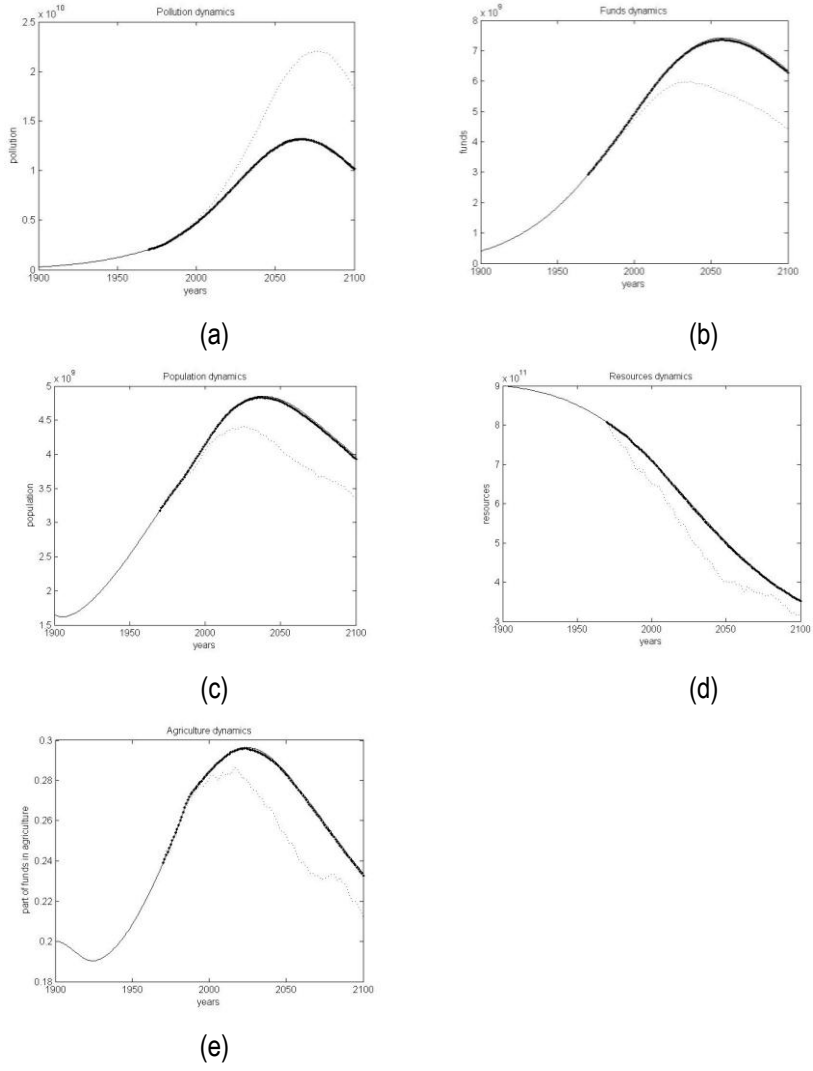


Fig. 2: Results of modeling (a) pollution dynamics; (b) funds dynamics; (c) population dynamics; (d) resources dynamics; (e) dynamics of capital investment in agriculture fraction.

One can see that the forecast is very close to the initial line, which reflects un noised function. It means that our model is stable to the 20% noise. In this case all functions converge and the relative root-mean-square deviation for every variable is equal:

- 0,31% for population
- 0,51% for funds
- 0,26% for agriculture
- 1,27% for pollution
- 0,46% for resources

Generally speaking, the Forrester model is very stability to the noise, which acts on the forecast period. Even when the noise level reaches 50% we have 69% convergent functions. This result also can be considered as the very good one.

3.2. Noise affects isolated variables on the stage of forecast

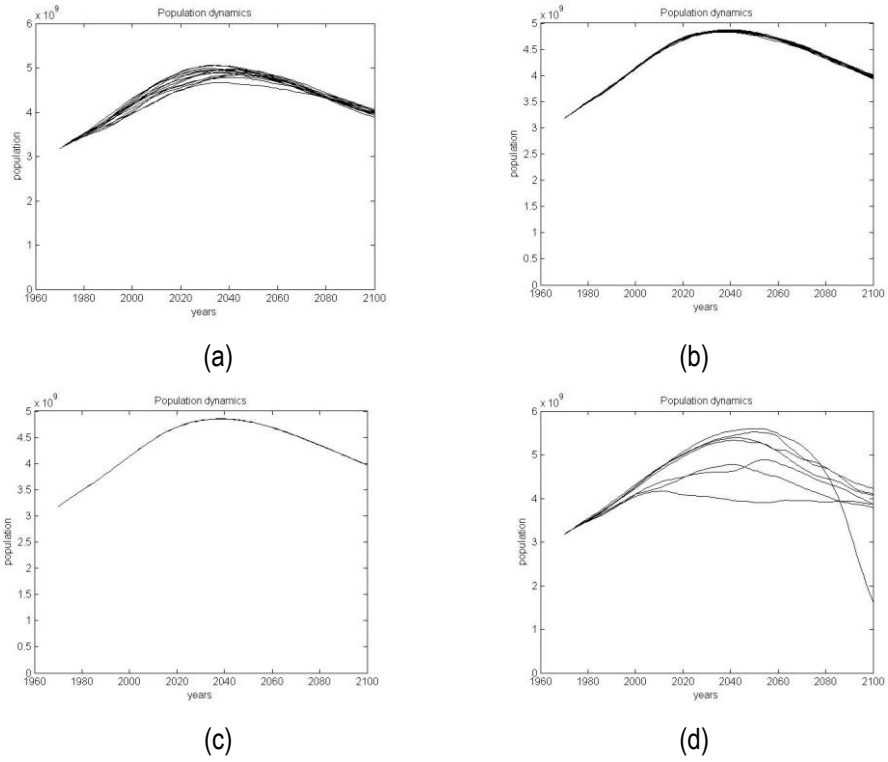


Fig. 3: Results of modeling pollution dynamics (a) noise affects only funds; (b) noise affects only agriculture; (c) noise affects only pollution; (d) noise affects only resources.

When noise affects only one variable the results of forecast change weakly. For that reason we used 50% noise to observe a significant difference. Our purpose is to reveal

the most influential variable. Figure 3 illustrates the population dynamics given the influence of different variables.

It is easy to see that resources are the most influential variable.

3.3. Noise affects initial values of all variables

As an example we consider the case with the 20% noise as we have done it in the section 3.1. Figure 4 presents the results of modeling for all macro-economical variables.

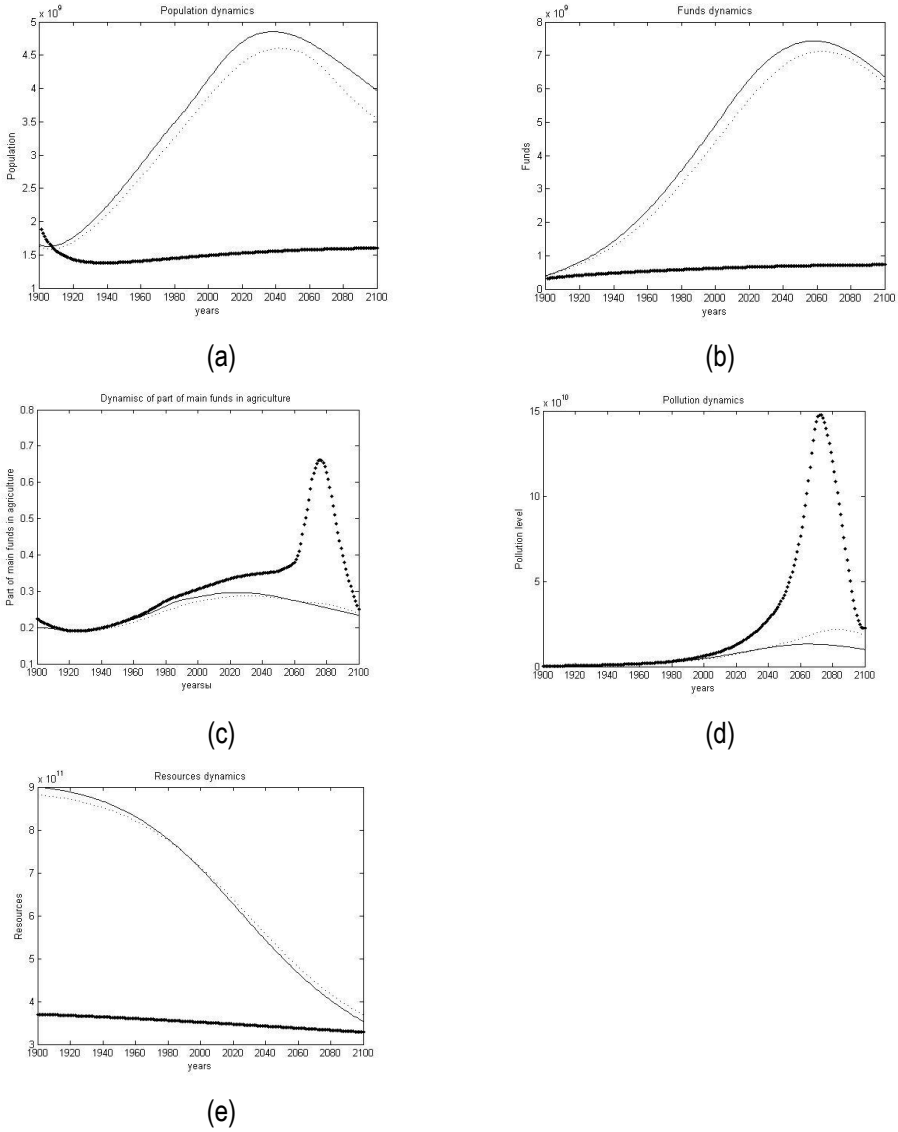


Fig. 4: Results of modeling (a) population dynamics; (b) funds dynamics; (c) agriculture dynamics; (d) pollution dynamics; (e) resources.

There are 3 lines on the figure: thin uninterrupted line is the initial function, thick line is the forecast, and thin dotted line is the worst function.

Let us compare the Figure 2 and the Figure 4. One can see the essential difference between the noise influence on the stage of forecast and the noise influence on the initial data. Namely, the Forrester model is more stable to noise on the stage of forecast. The related roof-mean-square deviations are:

- 14,58% for pollution
- 24,29% for funds
- 11,41% for agriculture
- 52,74% for pollution
- 12,97% for resources

3.4 Dependence on the level of noise.

The experiments described in the sections 3.1 and 3.3 were completed with the 20% noise. We repeated these experiments with the 10% noise to reveal the dependence of results on the level of noise. Table 1 jointed together the results of both experiments.

Table 1. Related roof-mean-square deviation of results

	Forecast 10%	Forecast 20%	Initial data 10%	Initial data 20%
Population	0,22%	0,31%	7,45%	14,58%
Funds	0,35%	0,51%	13,92%	24,29%
Agriculture	0,18%	0,26%	5,21%	11,41%
Pollution	0,82%	1,27%	31,05%	52,74%
Resources	0,33%	0,46%	7,34%	12,97%

It is easy to see that when the level of noise increases in 2 times then the deviation of results for the case of forecast increases in 1.5 times and for the initial data in 2 times.

4. Conclusion

In the paper we have studied the stability of the classical Forrester model to additive noise. The experiments show that

- the model is completely stable that is all its functions (variables) converge for the level of noise less than 22%;

2/3 of all functions converge for the level of noise 50%; here noise affects all variables on the stage of forecast

- noise in initial data causes the essentially stronger effect then the same noise on the stage of forecast
- the most influential variable is resources; its changes provoke the strongest reaction of the model

In future we plan to continue our research in two directions:

- to get the theoretical assessments of noise influence on the model for the case of stable mode of model functionality
- to analyze the noise influence on Egorov's and Makhov's models.

Bibliography

[Egorov, 1980] Egorov, V., Kallistov N., Mitrofanov, N., Piontkovsky, V. *Mathematical model of global development: a critical analysis of models of nature*. - Gidrometeoizdat, 1980. - 192. (rus).

[Forrester, 2003] Forrester J. *World Dynamics*. - Moscow: AST, 2003. - 379 p. (rus)

[Makhov, 2010] Makhov S. Long-term trends and projections in terms of a new model of world dynamics / forecast and modeling the dynamics of the global crisis/ Ed. A. Akayev, A. Korotayev, G. Malinetskii / Future Russia. - M.: LCI, 2010. - P. 262 - 276. (rus)

[Meadows, 2007] Meadows D., Randers J. *The Limits to Growth. 30 years later*. -M.: ICC "Akademkniga", 2007. - 342 p. (rus)

[Petrov, 2006] Petrov I., Lobanov A. "Lectures in Computational Mathematics." - M: The Internet University of Information Technology; BINOM. Laboratory of Knowledge, 2006. - 523 p. (rus)

Authors' Information

Olga Proncheva *– B.Sc. in economy and app. math., the Russian Presidential Academy of national economy and public administration; Prosp. Vernadskogo 82, bld. 1, Moscow, 119571, Russia; Moscow Institute of Physics and Technology (State University); Institutskii per 9., Dolgoprudny, Moscow Region, 141700, Russia*

e-mail: olga.proncheva@gmail.com

Major Fields of Scientific Research: mathematical modelling, world economy

Mikhail Alexandrov *– Professor of the Russian Presidential Academy of national economy and public administration; Prosp. Vernadskogo 82, bld. 1, Moscow, 119571, Russia; fLexSem Research Group, Autonomous University of Barcelona, 08193 Bellaterra (Barcelona), Spain;*
e-mail: MAlexandrov@mail.ru

Major Fields of Scientific Research: data mining, text mining, mathematical modelling

Sergey Makhov *– Asoc. Prof., PhD, Keldysh Institute of Applied Mathematics, Miusskaya sq., 4, Moscow, 125047, Russia;*

e-mail: s_makhov@mail.ru

Major Fields of Scientific Research: mathematical modelling, sinergetics, data mining.

INTEGRATED ENVIRONMENT FOR STORING AND HANDLING INFORMATION IN TASKS OF INDUCTIVE MODELLING FOR BUSINESS INTELLIGENCE SYSTEMS

Nataliya Shcherbakova, Volodymyr Stepashko

***Abstract**: Inductive modelling tools are widely used for solving problems of analysing economical, ecological, and other processes. Development of business intelligence systems based on inductive modelling algorithms for analysis, modelling, forecasting, classification, and clustering of complex processes is very promising.*

When solving real tasks of model construction from statistical data, the question of storage of and providing effective access to the information arises. At the stage of input data processing there are typical difficulties with processing data in different formats as well as containing omissions and untypically small or big values etc. From the other side, the question of output information storage exists like determination of structure and parameters of models, estimation of precision and validity, plots and diagrams drawing etc. This would allow structuring input data of different types and using the information already existing in database and also provide the storage of complete information on experiments and results of calculations.

To solve such kind of problems, the integrated environment for storing and handling information is developed. Architecture of the environment is offered giving the possibilities to manipulate present information freely using relational database containing only metadata and storing input statistical data and output results of calculations.

***Keywords**: integrated environment, handling and storing information, inductive modeling, GMDH-algorithms, Business Intelligence*

***ACM Classification Keywords**: H.2.8 Data Base Application – Data Mining*

Introduction

The number of companies which use business intelligence systems in their work is growing every year, so the development of such systems is continued; it is increasing constantly the number of systems, their functionality and technologies used for data processing and direct data analysis. Business intelligence systems are usually focused on

a specific task or function of a firm, such as analysis and forecast of sales, financial services, forecasting and risk analysis, trend analysis etc. Characteristic features of up-to-date BI systems are modularity, distributed architecture, the most common support and maintaining standards in web. At present, the development of systems aimed at business data analysis using OLAP, data mining and other. Growing range of algorithms is based on machine learning algorithms. Autonomous data mining tools are often included in other business analysis tools such as expanding database.

Analysis of algorithms used in business intelligence solutions

Business intelligence [Businessdictionary] refers to computer-based techniques used in spotting, digging-out, and analysing business data, such as sales revenue by products or departments or associated costs and incomes. But broader business intelligence can be characterized: firstly as process of converting data into information and knowledge for business decision support, secondly as information technology for data saving, information consolidating and guaranteeing access business users to knowledge, thirdly knowledge business gained as a result of data analysis and consolidation of information [Power, 2008]. Business intelligence solutions are widely used and have very diverse architecture, so there is no single specification of what those systems should be.

Objectives of a business intelligence exercise include understanding of a firm's internal and external strengths and weaknesses, understanding the relationship between different data for better decision making, detection of opportunities for innovation, and cost reduction and optimal deployment of resources.

For the past 10 years, the content and name of information-analytical systems have changed from information systems for manager, to decision support systems and business intelligence systems, second and third generation now. Business intelligence 2.0 is a new tools and software for business intelligence, beginning in the middle of 2000s, which enable, among other things, dynamic querying of real-time, web-based approached to data, as opposed to the proprietary querying tools that had characterized previous business intelligence software [Wikipedia]. Business Intelligence 3.0 is a term that refers to new tools and software for business intelligence, which enable contextual discovery and more collaborative decision making [Wikipedia].

Business intelligence tools is typically divided into the following categories: spreadsheets, reporting and querying software, OLAP, digital dashboards, decision engineering, process mining, business performance management, local information systems [Wikipedia]. Consider each of these groups in more detail.

Data mining is the process of extracting patterns from data; it is becoming an increasingly important tool to transform this data into information. It is commonly used in a wide range of profiling practices, such as marketing, surveillance, fraud detection and scientific discovery. DM can be used to uncover patterns in data but is often carried out only on samples of data. The mining process will be ineffective if the samples are not a good representation of the larger body of data. DM cannot discover patterns that may be present in a larger data body if those patterns are not present in a sample being "mined". Inability to find patterns may become a cause for some disputes between customers and service providers. Therefore data mining is not a proof fool but may be useful if sufficiently representative data samples are collected. The discovery of a particular pattern in a particular data set does not necessarily mean that a pattern is found elsewhere in the larger data from which that sample was drawn. An important part of the process is the verification and validation of patterns on other data samples.

Process mining is a process management technique that allow for the analysis of business processes based on event logs. The basic idea is to extract knowledge from event logs recorded by an information system. Process mining aims at improving this by providing techniques and tools for discovering process, control, data, organizational, and social structures from event logs. Process mining techniques are often used when no formal description of the process can be obtained by other means, or when the quality of an existing documentation is questionable. For example, the audit trails of a workflow management system, the transaction logs of an enterprise resource planning system, and the electronic patient records in a hospital can be used to discover models describing processes, organizations, and products. Moreover, such event logs can also be used to compare event logs with some a priori model to see whether the observed reality conforms to some prescriptive or descriptive model.

Business performance management is a set of management and analytic processes that enable the performance of an organization to be managed with a view to achieving one or more pre-selected goals.

Local information systems (LIS) are a niche form of information system - indeed they can be categorized as Business intelligence tools designed primarily to support geographic reporting. They also overlap with some capabilities of Geographic Information Systems although their primary function is the reporting of statistical data rather than the analysis of geospatial data. LIS also tend to offer some common Knowledge Management type functionality for storage and retrieval of unstructured data such as documents. They deliver functionality to load, store, analyse and present statistical data that has a strong geographic reference.

Table 1. Most popular data mining algorithms offered by three different BI solutions

	Pentaho [Pentaho]	Microsoft BI [Microsoft]	Oracle BI [Oracle]
Decision Tree (DT)	Classifiers	Predicting a discrete or continuous attribute, Finding groups of common items in transactions	Classification
Linear Regression (LR)	Classifiers	Predicting a continuous attribute	
Naive Bayes (NB)	Classifiers	Predicting a discrete attribute	Classification
Clustering	Clusterers	Predicting a discrete attribute, Finding groups of similar items	
Association Rules (AR)	Classifiers	Finding groups of common items in transactions	
Sequence Clustering (SC)	Forecasting	Predicting a sequence	
Time Series (TS)	Forecasting	Predicting a continuous attribute	
Neural Network (NN)	Forecasting	Predicting a continuous attribute	
Support Vector machine (SVM)	Classifiers		Classification and Regression
One Class Support Vector Machine (One-Class SVM)			Anomaly Detection
Generalized Linear Models (GLM)			Classification and Regression
Minimum Description Length (MDL)			Attribute Importance
Apriori (AP)	Association		Association
k-Means (KM)	Clusterers		Clustering
Orthogonal Partitioning Clustering (O-Cluster or OC)			Clustering
Nonnegative Matrix Factorization (NMF)			Feature Extraction

Business intelligence systems implement often classical data mining algorithms [Thomas, 2003]. Classification algorithms predict one or more discrete variables based on the other attributes in the dataset. Regression algorithms predict one or more continuous variables such as profit or loss on the bases of other attributes in the dataset. Segmentation algorithms divide data into groups or clusters of items that have similar properties. Association algorithms find correlations between different attributes in a dataset. The most common application of this kind of algorithm is for creating

association rules can be used in a market basket analysis. Sequence analysis algorithms summarize frequent sequences or episodes in data such as a Web path flow.

Furthermore, most business intelligence systems implement provisional data algorithms allowing eliminate gaps, specifically small or large data and convert output data to a required format.

Classical algorithms are mostly implemented as standard library for use in the systems being developed. We have examined main characteristics of the most popular business intelligence systems, namely the three different software packages: Pentaho, Microsoft BI, and Oracle BI. Table 1 shows the most known data mining algorithms offered by these three different BI solutions. It should be noted that Weka (Pentaho Data Mining) has near 100 classification schemes.

Prospects of inductive modelling algorithms usage in Business Intelligence solutions

Algorithms of inductive modelling are applicable for solving real-world modelling tasks in economical, ecological, and other processes [Ivakhnenko, 1985], [Ivakhnenko, 1982]. They are widely used in ill-defined systems developed for solving a specific problem. Below we examine a question of possibility of their use in business analytics; namely, which algorithms of inductive modelling and for what purposes can be used in business intelligence solutions.

The article [Ivakhnenko, 1968] published in 1968 by Prof. O.G. Ivakhnenko has marked the beginning of the new scientific direction called "inductive self-organizing of models from experimental data" or simply "inductive modelling" [Stepashko, 2008].

Group method of data handling (GMDH) is a family of inductive algorithms for computer-based mathematical modelling of multi-parametric datasets that features fully-automatic structural and parametric optimization of models. GMDH as personification of the inductive approach is an original method for constructing models from experimental data under uncertainty conditions. Models of optimal complexity obtained by this method reflect unknown laws of functioning of an object (process) information about which is implicitly contained in a data sample. To build models automatically, GMDH applies the principles of variants generation, nonterminal decisions and successive selection of the best models according to external criteria. These criteria are based on dividing the data set into two parts, where the tasks of parameter estimation and model checking are implemented in various subsets. GMDH algorithms are characterized by an inductive procedure that performs sorting-out of gradually complicated polynomial models and selecting the best solution using the external criteria.

A GMDH model with multiple inputs and one output is a subset of components of the base function:

$$Y = Y(x_1,\ldots,x_m) = a_0 + \sum_{i=1}^{m} a_i f_i(x_1,\ldots,x_m) \ , \qquad (1)$$

where *f* are elementary functions of different sets of inputs, *a* are coefficients and *m* is the number of the base function components.

To find the best solution, GMDH algorithms consider various component subsets of the base function (1) called partial models. Coefficients of these models are estimated by the least squares method. GMDH algorithm gradually increases the partial model complexity and finds an optimal model structure and parameters indicated by the minimum value of an external criterion. This process is called self-organization of models.

The most common base function used in GMDH is the Kolmogorov-Gabor polynomial:

$$Y(x_1,\ldots,x_m) = a_0 + \sum_{i=1}^{m} a_i x_i + \sum_{i=1}^{m}\sum_{j=i}^{m} a_{ij} x_i x_j + \sum_{i=1}^{m}\sum_{j=i}^{m}\sum_{k=j}^{m} a_{ijk} x_i x_j x_k + \ldots \qquad (2)$$

GMDH is used in such fields as data mining, knowledge discovery, prediction, complex systems modelling, optimization and pattern recognition. Application of GMDH algorithms for solving forecasting tasks allows their use in business intelligence systems along with other data mining algorithms.

Among GMDH algorithms that have more widespread gained, we can indicate the following [Ivachnenko, 2007]: Combinatorial (COMBI), Multilayered Iterative (MIA), GN, Objective System Analysis (OSA), Harmonical, Two-level (ARIMAD), Multiplicative-Additive (MAA), Objective Computer Clusterization (OCC), “Pointing Finger” (PF) Clusterization Algorithm, Analogues Complexing (AC), Harmonical Rediscretization, Algorithm on the base of multi-layered Theory of Statistical Decisions (MTSD), Group of Adaptive Models Evolution (GAME).

It should be noted that algorithms proposed by the most popular BI solutions are used for tasks of classification and clustering, mostly. For the numerical forecasting, developers offer: neural networks, linear regression and model trees. Such tools are less winning in forecasting tasks than GMDH algorithms. Requests for functionality of business intelligence system today lie in expanding their capacity of data mining. Other subsystems of business intelligence, such as data integration, import-export and reporting tools are developed very quickly. Thus the direction of business intelligence systems development using GMDH-based forecasting algorithms is very promising. Attention should also be drawn to the development of inductive modelling algorithms for classification and

clustering. Effective using such algorithms for modelling of complex systems is also an important argument for applying them in business intelligence systems.

Given the above we are developing our own system in which we use GMDH algorithms for modelling complex systems (processes). A necessity in accessible storage and drawing on scientific researches ripened already a long ago. An integrated environment for information storage would help to solve the existing problems, allowing structuring input data of different types and sources already existing in a data base, and also providing storage of complete information on experiments and results of calculations.

Integrated environment for storing and handling information

In [Shcherbakova, 2008] an architecture of integrated environment of handling and storing information in the tasks of inductive modelling was proposed, which allows freely manipulate the available information through building a layout environment which consists of relational database [Christopher, 1998], [Date, 2004], [Thomas, 2003] containing only meta-data and XML (PMML) storage, in which input data and results of calculations are stored [Graves, 2002].

Proposed layout of the environment is intended for solving problems of storage of input data and results. Designed architecture of the integrated environment for storing and handling information in the tasks of inductive modelling (Fig. 1) provides the opportunity to develop software. Modular system architecture makes it possible to expand its functionality.

The main requirements to the system is the ability to import (including primary processing) and export data, storing and handling existing information, storing output data with all information of calculation results, generate reports on the results. It should be noted that results of calculations are stored in the system in a standardized form that will allow generating strictly formal reports on the results of calculations.

Let us consider in more detail what information one needs to save in the system. Firstly, as already discussed above, these are input statistical data given to a single format and processed data with eliminated omissions and/or atypical values etc. Secondly there are basic functions, generated models, estimates of the parameters, criteria of quality models and best models. All the information is better to store in an XML storage. Auxiliary information such as data on the user, date and time use of files etc. it is better to store in a relational database.

Below we consider existing formats for saving predicting models and their use in our case.

Predictive Model Markup Language (PMML) is an XML dialect used to describe statistical models and models of data mining. Its main advantage is that PMML-compliant

applications can easily exchange models with other PMML-tools. The following classes of models can be kept using this markup language: associative rules, decision trees, center-based and distribution-based clustering, regression, general regression, neural networks, Bayes nets, sequences, text models, time series, rulesets, trees, support vectors.

for storage of predicting models, a scheme can be used in our case which describes a regression function. Regression functions are used to determine the relationship between the dependent variable (target area) and one or more independent variables. The term regression usually refers to the prediction of numeric values, hence the PMML element RegressionModel can also be used for classification. This is due to the fact that multiple regression equations can be combined in order to predict categorical values.

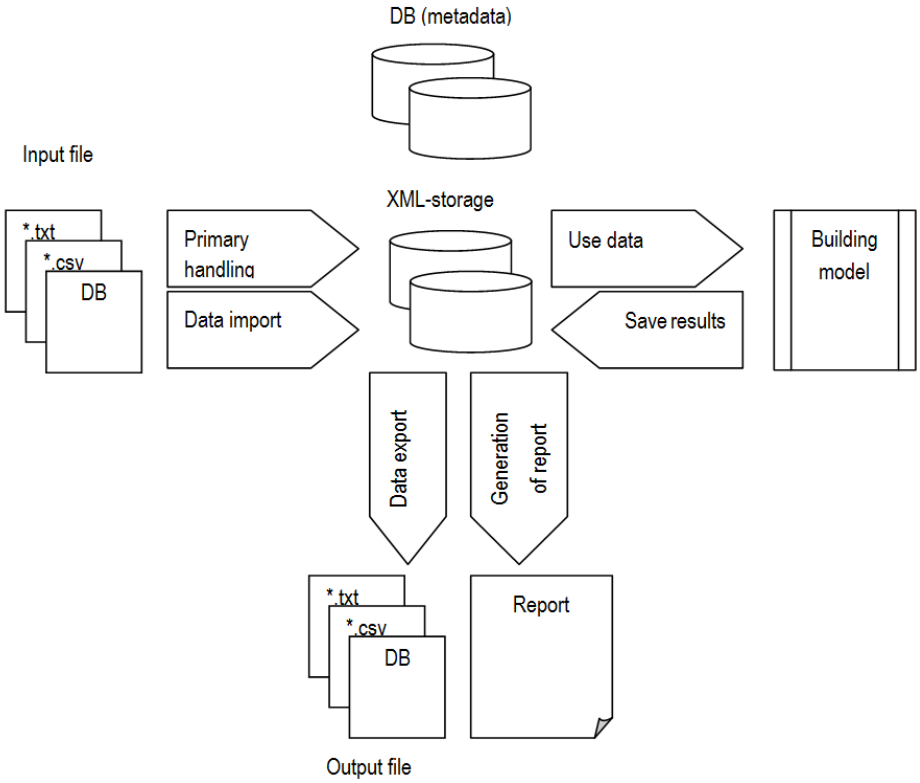


Fig. 1 Architecture of an integrated environment for information storing and handling

The offered integrated environment is intended for working out problems of storage of input statistical data and handling results. Developed architecture of the system of information storage in the tasks of inductive modelling gives a possibility to develop a software system that will allow freely manipulating the available information and adding new one.

Conclusion

This paper presents project of an integrated environment for storing and handling information in tasks of inductive modelling based on algorithms of the Group Method of Data Handling. Such type of system can be applied to solve some tasks of business intelligence, including forecasting, classification and clustering. Applying GMDH algorithms in business intelligence systems gives a particularly promising opportunity towards building complex models for business data analysis.

Bibliography

[Businessdictionary] http://www.businessdictionary.com/

[Power, 2008] Power D.J.: A Brief History of Decision Support Systems, version 4.0. DSSResources.COM. Retrieved, 2008.

[Wikipedia] http://en.wikipedia.org/wiki/Business_Intelligence_2.0/

[Wikipedia] http://en.wikipedia.org/wiki/Business_intelligence_3.0/

[Wikipedia] http://en.wikipedia.org/

[Thomas, 2003] Thomas K., Karely B.: Databases. Design, realization and accompaniment. Theory and practice. – M.: William, 1440 p, 2003.

[Pentaho] http://wiki.pentaho.com/ - Pentaho documentation.

[Oracle] http://msdn.microsoft.com/ - MSDN.

[Microsoft] http://docs.oracle.com/ - Oracle documentation.

[Ivakhnenko, 1985] Ivakhnenko A.G., Stepashko V.S.: Noise-immunity of modelling. – Kiev: Naukova Dumka, 216 p, 1985.

[Ivakhnenko, 1982] Ivakhnenko A.G.: Inductive method of self organization of models of complex systems. – Kiev: Naukova Dumka, 216 p, 1982.

[Ivakhnenko, 1968] Ivakhnenko O.G.: Group method of data handling - rival of method of stochastic approximation. //Automatic №3. – Kiev, pp 58-72, 1968.

[Stepashko, 2008] Stepashko V.S.: Theoretical aspects of GMDH as a method of inductive modelling. – Proceedings of the II International Conference on Inductive Modelling ICIM-2008, 15-19 September 2008, Kyiv, Ukraine. – Kyiv: IRTC ITS NANU, pp 9-16, 2008.

[Shcherbakova, 2008] Shcherbakova N. and Stepashko V. Integrated Environment for Information Handling and Storage in the Tasks of Inductive Modeling. – Proceedings of the II International Conference on Inductive Modelling ICIM-2008, 15-19 September 2008, Kyiv, Ukraine. – Kyiv: IRTC ITS NANU, pp 231-235, 2008.

[Ivakhnenko, 2007] http://www.gmdh.net/

[Christopher, 1998] Christopher J. Data: Introduction to databases systems. — K.: BHV, 608 p, 1998.

[Date, 2004] Date C.J.: An Introduction to Database Systems, Eighth Edition. – USA: Addison-Wesley, 1024 p, 2004.

[Thomas, 2003] Thomas K., Karely B.: Databases. Design, realization and accompaniment. Theory and practice. – M.: William, 1440 p, 2003.

[Graves, 2002] Graves M.: Designing XML Databases. M: «Vil'yams» Publishing house, 640 p, 2002.

http://en.wikipedia.org/wiki/Predictive_Model_Markup_Language/

Authors' Information

Nataliya Shchrebakova *– PhD student of IRTC ITS of NASU, P.A.: 40, Akademik Glushkov Prospect, Kyiv, Ukraine, 03680; e-mail: nataliya.shchrebakova@gmail.com*

Main Fields of Scientific Research: Information technologies of inductive modelling, Business Intelligence solutions

Volodymyr Stepashko *– Head of Department for Information Technologies of Inductive Modeling of IRTC ITS, Professor, Dr Sci, P.A.: 40, Akademik Glushkov Prospect, Kyiv, Ukraine, 03680; e-mail: stepashko@irtc.org.ua*

Main Fields of Scientific Research: Data analysis methods and systems, Knowledge discovery, Information technologies of inductive modelling, Group method of data handling (GMDH)

BI – SUPPORTING THE PROCESSES OF THE ORGANIZATION'S KNOWLEDGE MANAGEMENT

Justyna Stasieńko

Abstract. *The main goal of BI systems is to provide the access for the users to crucial information connected with the tools they use every day. It allows to take more relevant decisions, share knowledge with other people, cooperate within the whole organization and increase the company's gainings. The offered functionality includes either the scalable technology's platforms designed for workers in all management tiers.*

Keywords: *Business Intel's IIntelligence, Business Discovery, information, analysis, Qlickview*

ACM Classification Keywords: *K.6 Management of computing and information systems - K.6.0 General* economics

Introduction

The increase of the importance of information, which has become one of the most significant company resources, resulted in the growth of information systems' role in economy. These are especially Management Information Systems whose aim is to increase the competition through the use of IT in business strategy. The basic aim of Management Information Systems is to deliver as quickly as possible the information which is complex, reliable and appropriate as far as the form and the content are concerned and gives the answer for the current problem. The derivative aim is to enhance the management process in a given economic system in all management tiers.

The organizations are forced to process a vast amount of data because more and more data coming from the company's activity gives detailed information about consumers, new sales channels or extending the assortment. It turns out that using a worksheet for informing about business strategic decisions is not sufficient any more.

IT market offers many systems which constitute supporting tools for decision making processes. These are for example Support Decisions Systems, Executive Support Systems, Integrated systems – mainly ERP systems ect. Nowadays, they cannot always cope with all users' requirements. The excess of information, its arrangement,

inappropriate interpretation leading to mistakes have become the reason of creating Business Intelligence Systems (BI). BI systems are the generation of Management Information Systems (MIS) categorized according to the criteria of decision support level [Niedzielska, 2003]. The place of BI is presented by Fig.1. The role of BI is to support the processes of organization's knowledge management.

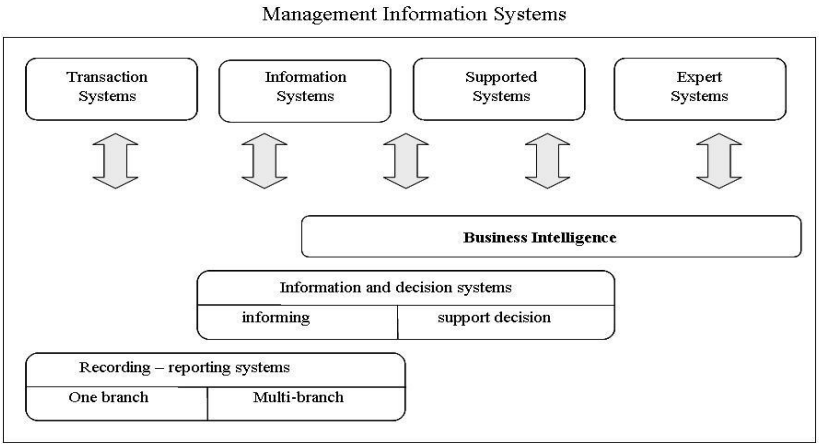


Fig.1. Management Information Systems.

Source: Niedzielska E.: Informatyka ekonomiczna. Wydawnictwo Akademii Ekonomicznej, Wrocław 2003, s.55.

BI systems

BI systems support decision making process and provide the access to up-to-date and reliable information. They help to find the answer for questions connected with the efficiency of the company's activity, explain the reason of the current situation and make plans for the future. They retrieve the data from many sources, usually in databases and then they analyse and process it. As a result BI is supposed to create the material which will constitute the base for drawing conclusions and rationalizing the process of making decisions.

In comparison with traditional systems, BI systems provide the quicker access, gathering and reliable information exploration. That is why, they use the advanced methods of data analysis such as mining, aggregation, crossing, casting and any other types of abstraction

levels of processes mapping in order to monitor the changing economic realities. They adjust the company to the current changes as quickly and precisely as possible and as a result make it more competitive. They analyse trends and relationships as well as their variation, consolidation and dissemination of information in the organization. They share data, models and computational formulas with other applications. They create typical reports and forms of presentation. They make it possible to make analyses ad hoc, predefine the set of questions and the access to the data and reports via the Internet portals. They contribute to the cost reduction of preparing reports and analyses. They are able to identify the chances and the risk, to show tendencies and consolidate the intuitive exploration of the main acts for the processes being realized. BI systems use the knowledge gathered in company's informative resources as well as the knowledge and experience of participants of the given process in order to understand the current processes' dynamics better. They can also combine the organization's strategy with operational activities, monitor the completion state of plans and prognoses and determine the costs of doing business. In comparison with any other systems they pass on the precise and up-to-date information to the users and define the economic processes in applications and automate them [Mańkowska, 2012].

BI systems may include the activity of the whole organization and make the detailed reports and analyses (Table 1).

The organizations start to notice the importance of budgeting and planning processes which are the following elements in converting the reporting and analytical systems into real BI systems. The process of reporting and analysing is important for each organization regardless of its size. The decisive factor which lead to the use of BI systems is not only the size of the organization but rather the number and variety of processes within its activity. In the organization with the expanded structure the time needed for gathering and processing the information is too long and that is why BI system is indispensable here. The implementation of BI system should be taken into consideration when the company's activity is complex. It is also helpful when there is no connection between the generated information and reports and the given strategy or the necessity of carrying plans and prognoses when the data concerning the future is incomplete. Another thing is a vast amount of databases and the tools from various contractors, lack of cohesion and unequivocal cause-an-effect-relationship between particular reports, lack of a quick and easy access to the processed information by managers as well as lack of satisfactory data concerning the analysed data.

Table 1. BI systems processes in particular activity areas of the organization

Area	Activity
Finances	- the current analysis of financial result - monitoring the condition of cash flow evaluation of financial ratio
Sale	- analysis of sales effect - prognosis of sales trends - analyses of products and profits profitability (analysis of the most and the least profitable products or the most and the least efficient workers) - marketing analyses sales verification
Logistics and buying	- estimation of stockhouse state - improvement of stockhouse economy - analysis of logistics processes - analysis of the efficiency of supply - contractors and collaborators' analysis
Production	- analysis of production costs - analysis of the use of production supply - analysis of production quality
Marketing	- analysis of market trends - providing the knowledge and experience gained while planning and introducing new products in the market - evaluation of the possibility of exerting an influence on the market (for example means on the advertisement) - evaluation of economic situation and development perspective
Human Resources	- analysis of work and register of workers' absence
CRM	- improving the relationship with clients - evaluation of some clients' profitability looking for key clients

Source: Own elaboration.

The organizations start to notice the importance of budgeting and planning processes which are the following elements in converting the reporting and analytical systems into real BI systems. The process of reporting and analysing is important for each organization regardless of its size. The decisive factor which lead to the use of BI systems is not only the size of the organization but rather the number and variety of processes within its activity. In the organization with the expanded structure the time needed for gathering and processing the information is too long and that is why BI system is indispensable here.

The implementation of BI system should be taken into consideration when the company's activity is complex. It is also helpful when there is no connection between the generated information and reports and the given strategy or the necessity of carrying plans and prognoses when the data concerning the future is incomplete. Another thing is a vast amount of databases and the tools from various contractors, lack of cohesion and unequivocal cause-an-effect-relationship between particular reports, lack of a quick and easy access to the processed information by managers as well as lack of satisfactory data concerning the analysed data.

The aim of implementing BI system on an advanced level is to support the process of making management decisions. The typical problems:

- to discover the reasons of the current business situation "what caused that something else happened?";
- identification process – "Which category should a particular case be numbered with?";
- prediction of further development of a given situation – "What may happen?"
- simulation of possible effects of making any management decisions;
- suggesting optimal options of the management decisions.

BI systems are supposed to enhance the organization's knowledge management in 3 levels presented in Table 2.

Initially the BI systems were restricted for supporting decisions in strategic and tactical area – while creating and developing products, while managing the finances as well as managing the process' efficiency. Currently, the properly implemented BI tools make it possible to deliver the information in any management level starting from the strategic one, through the tactical and daily management of the organization. The functionality of Business Intelligence solutions is not only restricted to a simple reporting but it can also support the management and realization of planning process, budgetary financing, reporting and the control of the achieved results. As far as the monitoring and formation of the companies' activity are concerned, they gain new possibilities while moving form time-consuming data collection and preparing the data and balance sheets to analysis of information which is currently made accessible through the advanced BI tools. BI applications of new generation are supposed to make accessible the information crucial to make decision on any operational level. BI tools play an important role in supporting the workers who are in direct touch with a customer.

Table 2. Tasks of Business Intelligence Systems

Management level	BI tasks
Operating	Analysis carried out ad hoc, information on current operations, finances, sales, collaboration with suppliers, customers, clients, etc.
Tactical	Fundamentals of decision making in marketing, sales, finance, capital management. Optimising future actions and modification of financial factors, technology in the implementation of strategic objectives.
Strategic	Precise setting of goals and tracking their implementation, to perform various comparative statements, conducting simulation development, forecasting future performance under certain assumptions.

Source: Own elaboration [Stasieńko, 2011].

BI system implemented in the operational level provides so called management information. It may occur in a form of charts, reports, tabular specifications etc. This information is created as a result of operations conducted on the data (exploration and aggregation). The management information is applicable in statistical methods, operational research and econometric models. It is typically used for costs monitoring and financial liquidity, analysis of net profit margin, analysis of manufacturing productivity and monitoring the condition of storage etc. BI systems have the advantage over classical reporting systems in which the analyses structure is "flat". The information is examined from only one point of view. BI systems are multi-dimensional structures. This feature allows to analyse at the same time the information from many points of view that can be changed dynamically. BI systems use also the methods of artificial intelligence. We can distinguish for example grouping through the analysis of the clusters, decision trees – sequential pattern analysis, classical pattern analysis – NN classifier and the classifiers based on decision-making within neural networks, inference system's [M. Flasiński, 2011].

BI systems can exist independently and take data from transactional systems (SCM and CRM) or they can constitute a part of them (a module).

BI in-memory

The growing number of the analysed data, the greater users' requirements and incapability of traditional BI platforms contributed to the change of attitude towards BI systems. There was a turning point in BI technology as well as in business users' awareness and preferences. There is a growing need for computing power. As a result there are innovative tools using in-memory bases called also Business Discovery. They are widely used as the most important analytical tools. The new technology makes it possible for BI users to use the flexibility, simplicity and service independence (self-service software conception), quickness and accuracy of analyses as well as the access to information in a form of attractive visualizations. QuickTech company with its QliView is a leader in this market. The application is elaborated on the pioneering analytical platform based on associative in-memory conception being developed from 1993. QlikView application is also available in Poland.

Apart from technological aspects, another thing enhancing the current changes is the business user's awareness. The above mentioned trend reflects the social changes connected with perceiving the software and application tools. Nowadays, IT consumers are accustomed to the quickness and service simplicity. IT tools in which various processes last longer are difficult to be accepted by the users. The users appreciate modern BI in-memory technologies where uploading and processing the data in operational memory, even if it is dispersed or gathered in different standards allows to make interactive real-time analyses and to receive the answers in split seconds. It is reflected in gradual transmission of BI tools from specialists to the business users.

Most Business Intelligence solutions cased on OLAP technology usually limit the analyses to several dimensions and as a result it forces the users to determine the questions they would ask in the future in the first phase of implementation. The traditional OLAP always downloads the data because it is impossible to perform an operation on the basis of others. The implementation and work in QlikView are different. The analyses are not restricted by any number of dimensions and the changes can be made a hoc very quickly. Such features of QlikView make it possible to use agile methods of projects management. The applications may be created evolutionally and the users are able to start using the effects of their activity and they can also take part in the process of their implementation. Another advantage is the possibility of adding new data to the application at any moment and from any source including for example text files, XML or Excel Worksheets. Such possibilities allow to modify the existing applications or creating new ones able to make any analyses in unusual situations. The analyses results are presented in form of

clear carts and balance sheets. Graphics can be arranged and changed from the point of view of various groups of workers and that is why the use of QlikView gives a lot of satisfaction. The managers monitor and control sales processes all the time so that they can detect even the smallest deviation and react as quickly as possible. QlikView is a complete business intelligence solution that consists of a data source integration module, analytic engine and user interface. QlikView is based on a patented architecture called Associative Query Logic (AQL) and is completely different from other OLAP tools.

Through AQL, QlikView eliminates the need for data cubes and data warehousing, replacing the cube structure with a patented data structure called a Data Cloud [Morejjon, 2012]. The Data Cloud does not contain any pre-aggregated data but instead builds non-redundant tables and keeps them in memory at all times. Queries are then created on the fly and are run against the Data Cloud's in-memory data store.

QlikView, thanks to the unique associative in-memory database technology, provides a quick insight into data and as a result makes it easier to take decisions for all business workers. It is all done thanks to one coherent analytical platform and it is quicker and cheaper. This platform was the basis for creating applications which enable making analyses of various types of data which were mentioned in [Stasieńko, 2010]. Developing technology creates new applications to analyse the data on a given subject. Last year there were created new applications connected with sport events like Olympic games and Euro 2012. The QlikView Global Games App captures the spirit of the 2012 Global Games. This application allows to analyse the event's stats, facts, trends, and trivia. It also enables to visualize, analyse, compare, and contrast game data by athletes, sports, countries and more. The data goes back to the very beginning of the modern games. This application contains historical medal winners data from 1896 through 2008 and daily updates of medal winners in 2012. The sports data, GDP, population and world record data are also available in this application for extended analysis. Fig.2. presents the amount of medals won during Olympic games by Polish competitors. Fig.3. shows a list of Polish Olympians of particular sport discipline and medals. The second chart depicts the number of medals won in particular sport disciplines. Fig.4. includes personal information about a given Olympian.

KICK-IT application and Qlik-IT Euro Football 2012 allow football fans to discover interesting facts connected with football. They make it possible to watch (data actualization took place in every 60 minutes) groups eliminations, quarter-finals, semi-finals and finals (Fig.5.), matches results, the best footballers etc. It is possible to check the information about the already played matches (fig.6.): the place, a group composition, shooters, changes and the received cards etc.

Fig.2. Application window, after choosing a particular country – Poland, which shows a number of medals won during Olympic games by Polish athletes
Source: Own elaboration.

Fig.3. A list of Polish Olympians of particular sport discipline and the obtained medals as well as the number of medals won in particular sport disciplines
Source: Own elaboration.

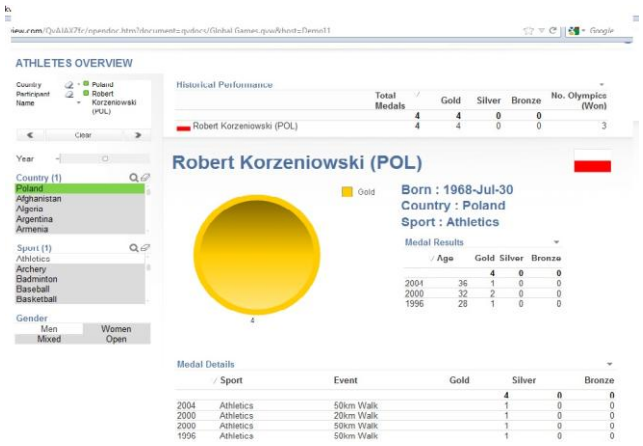


Fig.4. The information concerning the number of medals won during the World Championship by an athletic
Source: Own elaboration.

Fig.5. Data concerning the matches: winning, lost and drawn ones, score, and costumes worn by a given team.
Source: Own elaboration.

Fig.6. Detailed information on a particular match
Source: Own elaboration.

QlikView application is implemented in Polish organizations successfully. We can distinguish for example: banks, pharmaceutical company, companies connected with structural branch as: Tuplex, Saint Gobain Polska, Iglotech, Ever GRUPA, ESBANK Bank Spółdzielczy, EGIS Polska, Casinos Poland, BRE Bank SA, Biuro Podróży ITAKA, Antalis Polska. These companies chose QlikView solutions (BI In-memory) because of its computing Power, flexibility, low requirements as far as the equipment is concerned and its intuition. There are more and more systems on the Polish market which use in-memory technology. One of them is an integrated system Comarch CDN XL. "BI Start" module is used for creating reports and making analyses on the basis of data based on in-memory technology. As opposed to a "full" version of Business Intelligence packet, all the data generated by the reports is stored in computer memory. Among many other aspects, Comarch CDN XL BI Start has wide options of creating and manipulating the reports and

presenting them in more than 40 types of charts. What is more, it is able to manage the qualifications for activating the reports, to operate either domain accounts and SQL logins. It organizes reports in folders and makes it possible to create applications for sending e-mails with reports for particular users on the basis of a schedule (e.g. Every day they receive information about the storage and payment [5].

The market has been dominated so far by the tools based on traditional solutions (Table 3).

Table 3. BI tools in the particular IT companies

Software producer	**Product**
Oracle	Siebel Business Analytics Applications
	Hyperion System 9 BI+
SAS	Business Intelligence
SAP	BusinessObjects XI
IBM	Cognos 8 BI
Microsoft	Analysis Services
MicroStrategy	Dynamic Enterprise Dashboards
Pentaho	Open BI Suite
Information Builders	WebFOCUS Business Intelligence
TIBCO Spotfire	Enterprise Analytics
Sybase	InfoMaker
KXEN	IOLAP
SPSS	ShowCase

Source: Own elaboration.

Currently, the potentates in IT business (Microsoft, IBM, SAP, Oracle) try to include in-memory solutions in their offers. They usually create hybrid tools which combine classical BI and BI in-memory. On the other hand, BI products offered by these potentates lose their position of solutions treated as analytical platform standards in the organizations. There is going to take place a reshuffle among the main analytical solutions contractors because of such an intensive solutions evolution and the change of standards in work with business applications.

Conclusion

Nowadays, the attitude toward BI changes in many organizations. It is the end of era in which the business was run by intuition. The organizations, which are going to be successful, are the ones able to manage the information and use it in the decision making process.

BI systems give the possibility of increasing the competition on the market and taking proper and quick decisions.

BI systems help also to combine the initiative connected the enterprise's future with the branch of conducted business activity in order to understand and sort out the data in such a way that helps the managers to receive the crucial information in the proper time and as a result increase the enterprise's efficiency.

Business Intelligence systems deserve certainly credit in strategic management of the company. They constitute great tools in decision taking processes. As far as the analyses are concerned they can contain all areas of economic organization. Nevertheless, it is necessary to remember that analytical systems are risky and the success of their implementation is strictly connected with the use of their functionality by the users.

Bibliography

[Adamczewski, 2009] Adamczewski P.: Moduły BI w systemach klasy ERP. VII Krajowa Konferencja Bazy Danych: Aplikacje i Systemy, Zeszyty Naukowe Politechniki Śląskiej, seria INFORMATYKA,Wydawnictwo Politechniki Śląskiej, Gliwice 2009, s.49-58

[Buchnowska, 2008] Buchnowska D.:Analiza danych w systemach wspierających zarządzanie relacjami z klientami. Rozdział monografii: 'Bazy Danych: Rozwój metod i technologii', Kozielski S., Małysiak B., Kasprowski P., Mrozek D. (red.), WKŁ 2008

[Flasiński, 2011] Flasiński M.: Wstęp do sztucznej inteligencji. PWN, Warszawa 2011

[Januszewski, 2008] Januszewski A. Funkcjonalność Informatycznych Systemów Zarządzania, t.2 Systemy Business Intelligence, PWN, Warszawa, 2008

[Mańkowska, 2012] Mańkowska K.: Wykorzystanie systemów Business Intelligence w przedsiębiorstwie, http://www.pitwin.edu.pl/artykuly-naukowe/zarzdzanie/968-wykorzystanie-systemow-business-intelligence-w-przedsibiorstwie (06.2012)]
[Morejjon, 2012] Morejjon M.: Qliktech, IBMprovide newview of OLAP. http://www.crn.com/news/applications-os/18839582/qliktech-ibm-provide-new-view-of-olap.htm;jsessionid=dczRI7rxU7AMP3DdEVVM+g**.ecappj02
[Niedzielska, 2003] Niedzielska E.(red.): Informatyka ekonomiczna. Wydawnictwo Akademii Ekonomicznej we Wrocławiu, Wrocław 2003, s.55
[Nycz, 2008] Nycz M., Smok B. Busienss Intelligence w zarządzaniu. Materiały konferencyjne SWO, Katowice, 2008. http://www.swo.ae.katowice.pl/_pdf/421.pdf
[Olszak,2005] Olszak C. Wspomaganie decyzji w erze informacji i wiedzy. W: Systemy wspomagania organizacji. Praca zbiorowa pod redakcją H. Sroki i T. Porębskiej-Miąc. AE, Katowice, 2005, s.346-353
[Stasieńko, 2010] Stasieńko J.: BI in-memory – nowa jakoś systemów Business Intelligence. V Konferencj aNaukowa Information Systems in Management – Information Systems in Management IX – Business Intelligence and Knowledge Management, Warszawa 2011, s.88-98
[Stasieńko, 2011] Stasieńko J.: Business Discovery – A new dimension of Business Intelligence. Methods and Instruments of Artificial Intelligence, ITHEA, Rzeszów-Sofia, 2011. s. 141-148
[Surma 2009] Surma J. Business Intelligence – Systemy Wspomagania Decyzji Biznesowych, PWN, Warszawa, 2009

Netografia

[1] http://www.qlikview.com/
[2] http://www.qlikviewaddict.com/
[3] http://community.qlikview.com
[4] http://www.businessintelligence.pl
[5]http://www.comarch.pl/centrum-prasowe/aktualnosci/erp/comarch-cdn-xl-bi-start-nowosc-w-ofercie-comarch-erp]

Authors' Information

Justyna Stasieńko – *lecturer, The Institute of Technical Engineering, The Bronisław Markiewicz Higher State School of Technology and Economics, Czarnieckiego Street 16, 37-500 Jarosław, Poland; e-mail: justyna.stasienko@pwste.edu.pl*

Major Fields of Scientific Research: Management Information Systems, Business information technology

INTELLIGENT METHODS OF REVEALING FRAGMENTS IN BUSINESS PROCESSES

Nataliia Golian, Vira Golian, Olga Kalynychenko

Abstract: *The Effective methods of intelligent analysis of business processes, in particular, methods of revealing fragments of such processes are developed. Besides, analyzing information extracted from journals of registering events of a business process (BP) to formalize the real behavior of a BP is carried out. Such data analysis is especially important in those cases when the occurring sequence of events is registered, i.e. executives have an opportunity to make a decision about the order of further process implementation.*

Keywords: *business process, procedure, logical net, intelligent analysis.*

ACM Classification Keywords: I.*2 Artificial Intelligence – Knowledge Representation Formalisms and Methods*

Introduction

The term, which occurs the most frequently in this paper, is a "business process". According to literature it is orderly set of tasks that require one or more factors of production and generate result, which is focused on fulfilling customers' needs. Modeling and computer simulation have an increasingly important position amid tools are used by engineers and managers. This is the result of need to make quick and accurate decision in response to constantly changing environment. Furthermore, manufacturing systems are more complex than before. Through information technology that are applied in conjunction with software for modeling and simulation, issues difficult to solve due to the high complexity, can by deeply analysed. Modeling is to build a virtual model, which illustrates a real business process. Then, simulations are carried out on this model.

The purpose of this paper is an intelligent analysis of a business processes for formalizing its actual behavior.

The actuality of considering the structure and characteristics of business processes in the context of the present work is defined by the need for developing efficient methods of intelligent analysis of business processes, particularly, such processes.

At present there is a change-over to process management from a traditional functional one, which requires the formalization of existing business processes by constructing their hierarchical structure with the use of typical fragments of the processes. Hence, working out details of the structure and characteristics of business processes is a necessary condition of researching and developing methods of intelligent data analysis.

Constructing formal models of business processes requires considerable time and financial expanses as well as it is affected by a human factor, as it is often realized in concert with experts and executives, whose aims cannot coincide with the ones of the business processes under simulation and the enterprise common goals. This contradiction can lead to discrepancy between a real business process and its developed model. It can result in constructing inadequate models of business processes.

One of the main approaches to solving the given problem is realized on the basis of the methodology of intelligent analysis of business processes. The intelligent analysis is directed towards obtaining models of really implemented business processes on the basis of researching logbooks of events of such processes (files – logs).

Such data analysis is particularly important in those cases when an occurring sequence of events is being registered, however, the process is partially or completely non-formalized, i. e. executives have an opportunity to take decisions about the order of further process implementation, proceeding from the local information they have and in view of a BP. In fact, implicit relations between BP procedures are realized in such processes. The said relations are based upon the knowledge which is not involved in the process description, they can result in variations from its documentary behavior and at that are not practically identified on the basis of existing approaches in the field of intelligent analysis of business processes.

At present algorithms of constructing models of business processes on the basis of analyzing logbooks of events are developed [Agrawal, 1998; Aalst, 2003; Medeiros, 2003; Aalst, 2004]. At the same time the developed approaches do not allow to completely solve the problem of revealing implicit relations and, with their help, constructions of implicit choice in the structure of BPs.

Importance of simulation modeling and value analysis in the present-day world

One of the main trends that have been outlined lately in the field of information technologies is the increased interest for methodological and technological problems of using simulation modeling in the practice of researching and designing sophisticated systems in various application areas that is due to the following causes:

- by extending of the area of applications of simulation modeling first of all owing to such nontraditional directions as BPs, marketing, logistics, financial management, socio-economic processes, etc.
- by increasing of the level of manufacturability of simulation systems due to render features: graphical interface, animation as well as Case – technologies. Lately the unified language of visual representation of models of program systems – UML (Unified Modeling Language), developed by the famous American experts in the field of object – oriented programming Gradi Buch, Jimmy Rumbach and Ivar Jacobson has been widespread.
- By mass use of Internet – technologies for both support distance learning processes and realizing simulation experiments on the basis of modern network technologies. At present the websites of such famous experts in the field of simulation modeling as R. Sergeant, O. Balchi, R. Fujimoto and others are accessible to the broad researching public. Through websites one can get materials of such an important event as Winter Simulation Conference which is actually held by the International community of simulators. The Russian simulation portal gpss.ru regularly publishes the "hottest" information from foreign and Russian practice of simulation modeling.
- By development of opportunities of designing and studying sophisticated systems on the base of so-called models of virtual reality.

The reports submitted at the last three Winter Simulation Conferences (2000-2002) are evidence of the increased interest and demand for simulation modeling systems.

The first all – Russian theoretical and practical conference UMMOD-2003 was held in the city of Saint-Petersburgh (Russian Federation). Over 30 reports were devoted to research in different application areas (space manufacturing, logistics, medicine, ship building, transport, etc.). The reports presented are evidence of the scale and high level of projects being developed in Russia at present on the basis of simulation modeling methods.

Main information about simulation modeling and value analysis

Simulation modeling is a method of study allowing to analyze a system without changing it. It is possible due to the system under investigation being replaced by a simulating system, the information obtained is characteristic of the system investigated. Speaking about analysis of a company's activity the method allows simulating the implementation of BPs in such a way as if it occurred in reality and getting a real estimation of the duration of each process.

Value analysis is an instrument designed to estimate product (service) costing. Implementing value analysis allows to estimate cost price through management of processes directed at manufacturing a product or rendering a service. It is in this that the method of value analysis differs from traditional financial methods of calculation of expenditures within which the company's activity is estimated on the basis of functional operations but not by specific products (services) provided to a customer. The following point underlines value analysis: to produce a product (service) requires implementing a number of processes, consuming specified resources. Expenses on implementation of a process are calculated by transferring costs of resources on costs of process steps. The amount of expenditures on implementing all the processes with definite amendments constitutes product costing. If the traditional methods calculate expenses on a certain activity status by cost categories, value analysis shows costs of implementing all the process steps. Thus, the methods of value analysis allows to calculate expenses on manufacturing products (rendering services) most accurately as well as they present information for analyzing processes and their improvement.

Value analysis and simulation modeling stages include developing a model of processes, giving time parameters of finite (non-decomposed) processes. The resources are subdivided into time and material ones. The cost of a time resource is carried over to the cost of a material resource proportionally to that time which the resource takes to implement the process. The cost of a material resource is carried over to proportionally the quantity of repetitious of a process, purposes of resources on processes, performing a simulation of implementing the processes.

Analysis of the main trends in the area of present-day simulation modeling

One of the main trends in the area of developing and introducing modern systems of BP management is using simulation modeling as their integral part. Simulation systems, embedded into BP management, provide performing of such important tasks as project control, resource planning, control of business rules, investment forecast on the basis of analysis according to the scheme "what-if", training in new/reorganized BPs, script planning for emergency conditions. At that generally accepted is the point of view that simulation modeling must accompany BPs from the very starting stage of their formation, development and introduction.

Examples of such famous modeling systems as SIMUL8 (SIMUL8 Corporation), AutoMod (Brooks Automation), ProModel (ProModel Corporation) and WITNESS (Lanner Group Incorporated) can serve a confirmation of the mentioned trend.

SIMUL8 Corporation develops supplies to the market and supports simulation modeling systems oriented on performing tasks of business, government, education and organizations that face problems of managing flows of orders, clients, transport or production. Within a comparatively short period of time since the date of its foundation in 1994 the corporation has managed to include very many solid firms, namely: IBM, Bell Laboratories, Motorola, Ford Motor Company, Boeing Aircraft, British Airways, Virgin Atlantic, Hewlett-Packard Corporation, USA Air Force, British Steel and Nissan Motors in the list of its customers.

The systems of modeling AutoMod, ProModel and WITNESS have also found wide application in various application areas (manufacture, business, storehouses, logistics, transport, production of pharmaceuticals, reengineering in business) and have found such firms as General Electric, Intel, Siemens, Nokia, Motorola etc. as serious customers.

Two directions are distinguished in the VR-area of simulation modeling: the first one is related to videogame industry and the second direction is in the first place related to problems of researching and analyzing industrial processes on the basis of e-Manufacturing concept which received its development in the automobile manufacture industry of Germany in the late 1990s.

The main aim of using e-Manufacturing is progressing to such stage of modeling objects and processes which provides a possibility of studying in detail and optimizing all aspects of any productive process before starting its initiation.

It is natural that transferring to the e-Manufacturing regime especially of large-scale enterprises can only be gradual. Such concerns as Mercedes-Benz PkW, Opel, BMW, Audi, Toyota, Airbus (when manufacturing the airbus A-380 in Hamburg) are planning to introduce the idea of e-Manufacturing.

As a whole the concept of e-Manufacturing can be represented by the formula “Simulation + Virtual Reality”. Implementing the concept of e-Manufacturing requires having the following software support tools:

- Storage of text and graphic data represented in different formats;
- Simulation modeling of systems and processes investigated.
- Visualization of results of modeling by VR methods.

Only two firms – DELMIA and Technomatix – are ready to offer complete sets of mutually compatible software products for supporting e-Manufacturing concept at the European market of software products. The core of each system is a special data bank which represents three basic structures of industrial purpose – PPR (Product, Process, Resources). This bank is called an e-Manufacturing Server and PPR Hub by Technomatix

and DELMIA respectively. Technomatix uses EM-Plant and DELMIA employs QUEST as simulators.

The Magdeburg Institute of Organization and Automatization of Industrial Production named after Fraunhofer – IIF –carries out a large scope of work on use of simulation modeling in industry (particularly, the EM-Plant systems) as well as on creating VR-models.

The institute has accumulated a considerable experience of developing both VRMI-models for representing industrial processes (including simulation modeling -based one) and special VR-models fully dipping a customer into virtual space. The latter allows him/her to do work on designing and testing machines and equipment. VR-models are used to instruct and train people – operators mastering new operations.

Thanks of the achievements of the institute in the area of developing and constructing virtual models a decision has been taken to construct the Virtual Development and Training Centre (VDTC) in Magdeburg. It is planned that the enterprises and organizations will be proposed a wide spectrum of services on bringing facilities and technology of developing VR-models to a commercial level, training of personnel operating sophisticated equipment, implementing repair and preventive work.

Problem statement

In really functioning BPs one can distinguish two types of choice of procedures which represent different kinds of relations between them: explicit and implicit ones.

The explicit relations [Desel, 2005] represent cause – and – effect relations between procedures. At that such procedures are usually available in pairs in a logbook of BP events. Example: For starting the implementation of procedure P2 it is necessary that procedure P1be completed.

Implicit relations represent indirect cause – and – effect relations between procedures. The interrelation between P1 and P2 is not seen immediately from the logbook of events. For instance, procedures P1 and P2 can be interconnected through a sequence of procedures <P3, P5>. It is this that defines the importance of formalizing typical constructions of implicit choice.

Hence, the problem is to obtain formal algebraic logical models of constructions of implicit choice on the base of analyzing the main peculiarities of typical sequences of procedures with implicit choice in BPs as well as errors of revealing such fragments with existing algorithms.

Implicit relations in BPs represent indirect cause – and – effect relations between procedures and possess the features of connectivity, reachability and do not have the feature of a sequence. In the case of implicit relations between the considered procedures of a BP there exists a chain of other procedures (indirect relation), which makes the revelation of such fragments more difficult.

All the above mentioned defines the importance of formalizing implicit relations between procedures.

The paper's task consists in obtaining formal models of implicit relations between BP procedures that would possess the following peculiarities:

- Representing parallel and sequential implementation of the current fragment of a BP and its other subprocesses;
- Covering the necessary and sufficient set of features of implicit relations allowing to single them out on the basis of analyzing the sequence of procedures of the BP implemented.

Algebraic logical models of implicit choice constructions

Define formally explicit and implicit relations between BP procedures by finite predicate algebra. Let us enter variables $x_{1,}, x_2,..., x_n,,$ denoting the states of procedures $P_{1,_2} P_{2,},..., P_n$. These variables are given on some finite set of possible values of procedure states. For example, $x_1 \in \{a, b, c\},$, where $x_1 = a$ means "procedure p_1 has not been implemented", $x_1 = b -$ "procedure p_1 is being implemented", $x_1 = c$ means "procedure p_1 has been implemented".

Let us enter predicates $x_{1,}, x_2,..., x_n,,$ $L_{1,}, L_2,..., L_k,,$ denoting pair wise relations (if they exists) between procedures $P_{1,_2} P_{2,},..., P_n$. For the example mentioned above the relation between procedures P_1 and P_2 will be described by a predicate $l_1 (x_{1,} x_2)$.

The constructions implementing implicit choice are characterized by a contradiction between a choice of some alternatives and the necessity of synchronizing chosen actions with those already being implemented within a BP. In other words, the implicit choice situation is characterized by a combination of synchronization and choice constructions, as shown in Fig.1.

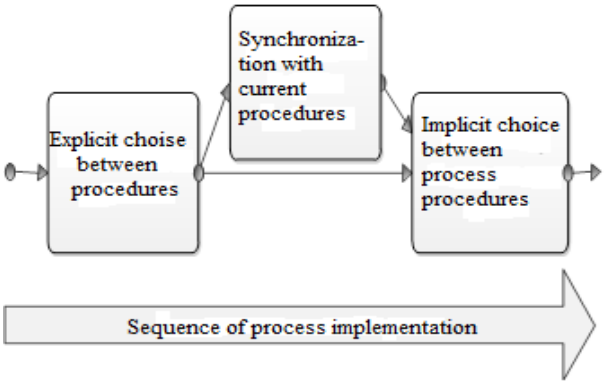


Fig. 1. Implicit choice situation between business process procedures

As it is seen from the figure, the implicit choice is defined by a choice of such and such procedures at the previous stages of a process and is implemented after synchronizing the results of the earlier choice with current procedures. Such synchronization is necessary so that all input conditions for final procedures, between which an implicit choice is made in the course of implementing a BP, may be satisfied. The problem of revealing implicit choice structures in intelligent analysis problems of BPs is defined by the fact that the existing algorithms whose mathematical base is formed by Petri nets, particularly WF-nets [Li, 2003] as their extension, usually cannot process such constructions.

Fig.2 shows a situation, according to which the final result of implementing the current BP fragment depends on an implicit choice between procedures P_4 and P_5.

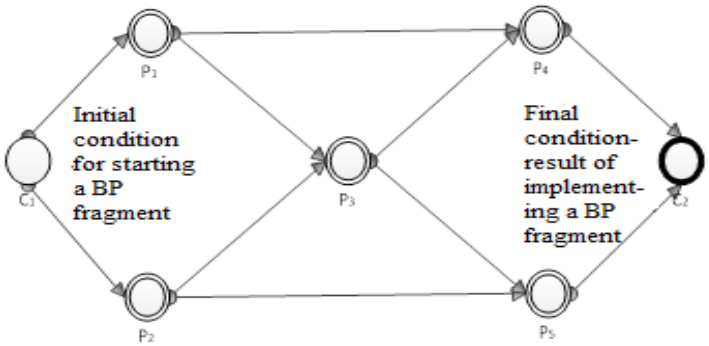


Fig. 2. Situation of implicit choice between P4 and P5

Implementing the given fragment starts in the case if initial condition C_1 is satisfied. Then an explicit choice between procedures P_1 and P_2 is implemented. The result of implementing procedures P_1 and P_2 must be synchronized with the result of procedure P_3, following which a choice between procedures P_4 and P_5 is made. As choosing one of the mentioned procedures is defined by a choice of P_1 and P_2 at an earlier stage of implementation, we obtain that the choice between P_4 and P_5 is implicit.

The operation of such a logical net will consist in a cyclic check of a set of all the relations between its BP procedures which exists in a modeled BP, which at each step of calculations will take the form of generation of a set of procedures that must be implemented to continue a BP.

Enter variables $x_{1,}, x_2, ..., x_5$, denoting the states of procedures $P_{1,_2} P_{2,}, ..., P_5$. These variables can take values from the set $\{0,1\}$, which means implementation or non-fulfillment of the procedures respectively. To construct a logic net let us find a system of binary predicates $L_{1,_2} L_{2,}, ..., L_k$, that describes the logic of executing a fragment of BPs, shown in Fig.2. With this in mind enter temporary variable t, containing information about explicit and implicit choices of procedures. The variable t can take values from the set $\{0,1,2,3\}$, denoting implementation of procedures $P_{1,_2} P_{2,}, P_1$ and $P_{3,_2} P_{2,} and P_3$.

The system of predicates describing a logical net of the first typical implicit choice situation takes the form:

$$\begin{cases} L_1(x_1, x_3), \\ L_2(x_{1,} t,), \\ L_3(x_{2,} x_3), \\ L_4(x_2, t), \\ L_5(x_3, t), \\ L_6(t, x_4), \\ L_7(t, x_5). \end{cases} \qquad (1)$$

The appropriate logic net is shown in Fig. 3.

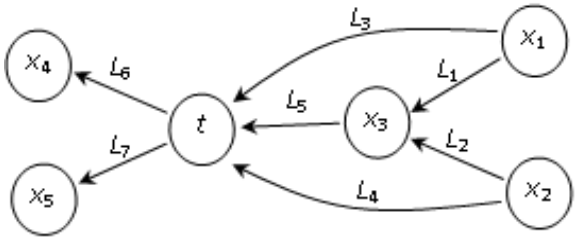


Fig. 3. Logical net of the first typical situation

The initial nodes of logic net are variables x_1, x_2 and the final ones are variables x_4, x_5.

At the beginning of operation of a logical net all its variables, modeling BP – procedures, have the value of 0 – none of the procedures is not implemented. In the course of operation of the logical net, i. e. calculating a system of binary predicates which corresponds to this net, all or some variables *x1, x2,…, x5* of the logical net take the values of 1 – all the procedures have been implemented. The variable t will take the value, corresponding to a sequence of implementing procedures, which is realized in a BP.

Consider an example of operation of a constructed logical net in steps:

Step 0 (beginning of operation of a net): x1=0,…,x5=0, tis not defined.

Step 1: x1=1, t=2, the remaining nodes have no changes.

Step 2, halfstep 1 (net state) :x3=1.

Step 3 (completion of operation of a net) :x4=1.

It is obvious that, if the implementation of a procedure P2 is the initial action in the net, then completion of the net operation will result in x5=1. The second typical situation of implicit choice of a procedure sequence is shown in Fig. 4.

In the given situation there are two implicit choices – between procedures P_4 *and* P_5, as well as between procedures P_3 *and* P_5.

Sequence of this fragment is as follows. The initial condition C1 is an input for a single procedure P1. As a result, its performance can be parallel (in any order) performed the procedure P2 and P3 respectively. Further, there are several implementation options.

Option 1. In that case, if the procedure is executed before the procedure P2 P3 is the procedure P3, P4 and P5 can be performed in any order, independently of each other. Then you can choose from two parallel branches - either performed the procedure P6, or both of the procedure, P6 and P7.

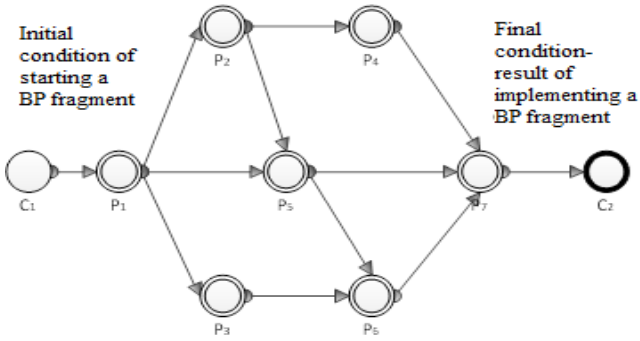


Fig. 4. Situation of implicit choice between procedures P_4 and P_5, P_3 and P_5.

Option 2. If the procedure is executed before the procedure P3 P2, then it becomes a further order of the hard-coded: P2 should be executed after the P4 and P5, P5 is performed after the P6, and then - P7.

Thus, the set of possible sequences of the procedures can be represented as tuples of the following:

{<P1, P2, P5, P6, P7>, <P1, P2, P3, P4, P6, P7>, <P1, P3, P2, P6, P4, P7>,

<P1, P2, P4, P3, P6, P7>, <P1, P3, P6, P2, P4, P7>}.

The problem of identifying the fragment is as follows. Existing data mining algorithms do not recognize the connection between the procedures P1 and P5, P5 and P7, which can lead to delays and deadlocks in process models, in particular, if a pair of parallel processes P4 and P5 will be performed only one procedure P5. The direction of solving this problem is to identify the cause - effect relationship between the procedures P1 and P5, P5 and P7 on the basis of analysis of event log.

We construct algebraic-logical model of a typical situation, an implicit second choice of the sequence of procedures similar to the first type situation. We introduce the variables x1, x2, ..., x7, denoting the state of processes P1, ..., P7. These variables can take values from {0, 1}, respectively, which means the performance or failure to comply with procedures. To construct a logical network will find the system binary predicate L1, L2, ..., Lk, which describes the execution logic of business process fragment shown in Fig. 2.4. For this purpose, we introduce an intermediate variable t, which contains information about the sequence of explicit and implicit election procedures. The variable t can take values from {0, 1} by the formula: t = 0, if the procedure is executed before the procedure P2 P3; t = 1, if the procedure is executed after the procedure P2 P3. The system of binary predicates, that describes a logical net of the second typical situation of implicit choice, consists of 12 predicates $L_{1_2}, L_2, ..., L_{12}$. . The appropriate logic is shown in Fig. 5.

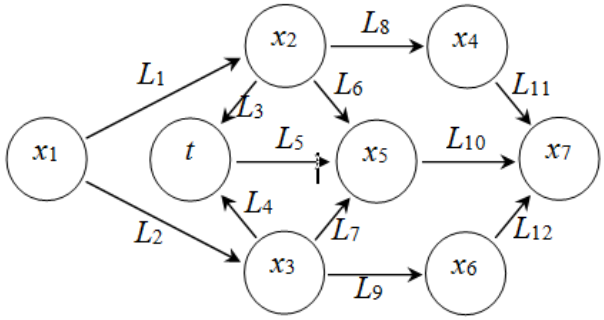


Fig. 5. Logical net of the second typical situation

The model of a BP in the form of a logical net, as shown above, generates at any moment of time a set of procedures that must be implemented to continue the BP. In fulfilling any next procedure the situation changes, which at once will result in the change of the state of a logical net and, respectively, the change of a set of procedures which must be realized to continue the BP.

Typical situation 3 of implicit choice of the sequence of procedures is shown in Fig. 6. There exists an implicit choice between procedures P3 and P5. The sequence of implementing the given fragment takes the following form. After realizing procedure P1 the nets operate practically in a sequential regime and do not have an advantage over the models of the same processes in the form of multi-place predicates.

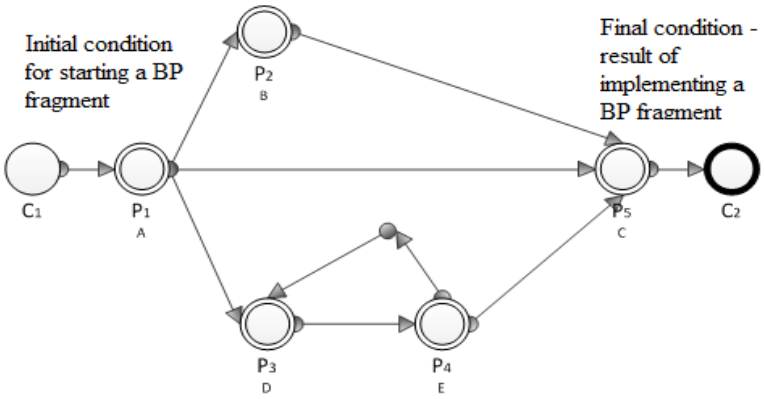


Fig.6. Situation of implicit choice between procedures P3 and P5

P5 can be executed after the loop P3 → P4, and the procedure P2.

Consequently, the possible sets of sequences of the procedures can be represented as tuples of the following:

{<P1, P2, P5>, <P1, P2, P3, P4, P5>,

<P1, P3, P2, P4, P5>, <P1, P3, P4, …, P3, P4, P2, P5>}.

The solution of the problem considered in this case involves finding the relationship between the procedures P1 and P5.

Construct a logical network of third typical situation an implicit choice of the sequence of procedures similar to those discussed above situations. We introduce the variables x1, x2, ..., x5, denoting the state of processes P1, ..., P5. We find the system binary predicate L1, L2, ..., Lk, which describes the execution logic of the movie business processes depicted in Fig. 2.6. For this purpose, we introduce an intermediate variable t, which contains information about the implicit choice between procedures, P3 and P5. The variable t can take values from {0, 1} by the formula: t = 0, if the procedure is executed before the procedure P3 P5; t = 1, if the procedure is executed after the procedure P3 P5.

The system of binary predicate that describes the logical network is a typical situation, an implicit second option consists of seven predicates L1, L2, ..., L7. The corresponding logical network is shown in Fig.7.

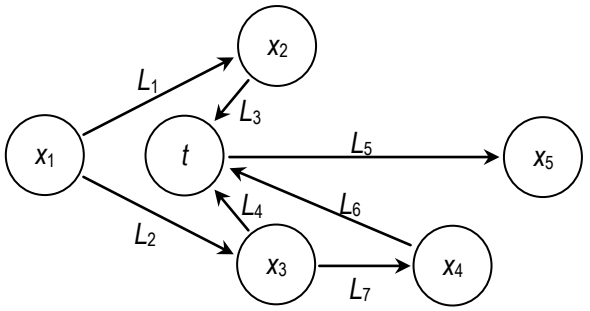


Fig. 7. Logical network of third typical situation

Consider the example of the logical network built on the cycles:

0 cycle (beginning of the network): x1 = a, the remaining nodes unchanged.

1 cycle: x2 = 1, the remaining nodes unchanged.

2 cycle: x3 = 1, t = 0, the remaining nodes unchanged.

3 cycle: x4 = 1, the remaining nodes unchanged.

4 cycle (the end of the network): x5 = 1.

Development of predicate models of implicit relations between procedures

Developing the method of identifying implicit choice situations requires formalizing of the main features of such situations, namely, formalizing of implicit relations of various type, which requires developing models of representing implicit relations between procedures.

This subsection is devoted to the development of predicate models of representing implicit relations between BP procedures on the basis of the algebraic logical model of a generalized implicit choice construction. The given model combines four schemes of interaction of an implicit choice construction and other BP fragments. Therefore four types of implicit relations between procedures will be further considered and formalized in the form of predicate models.

Realization of implicit choice models is based on a predicate model of representing an indirect relation between procedures in the logbook of BP events. Consider and conclusively formalize the implicit relations of all four types. Implicit relation of type 1: outputs of an outside subprocess – inputs of an analyzed fragment.

The given type of a relation is based upon the interaction of the form: output of an outside subprocess – inputs of final procedure Pi and the intermediate chain of procedures P2, … ,Pn (Fig. 8).

In accord with the given scheme of interaction let us formulate a set of conditions defining the relation of the given type:

- There is no explicit relation between initial P1 and final Pn procedures of the fragments under study.
- The initial procedure of fragment P1 in the model, obtained on the basis of analyzing a log of operations, has only one output that is an input of the procedure $P_j, j = \overline{2, n-1}$
- The procedure Pj has a second input which is common for the final procedure Pn.
- The input, common for procedures Pj and Pn, is the result of operation of procedure Pk that falls into a different situation of the given BP or a different subprocess. It should be noted that splitting in situations and subprocesses is included in the structure of a logbook of events, as there usually exists the column “Situation code”
- Procedure P5, outside relative to the analyzed fragment (generalizing the whole outside subprocess), is not related to initial procedure P1 in input-output;

- There is an indirect relation between procedures P1 and Pn.

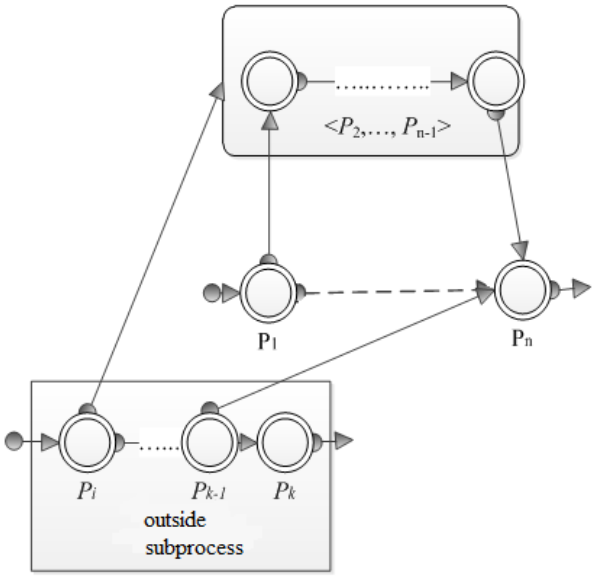


Fig.8. Implicit relation of type 1: outputs of the outside subprocess – inputs of the fragment under study

There is a type 1 implicit relation in meeting the considered six conditions between procedures P1 and Pn.

Summing up the above-mentioned, one can say that the given relation allows identifying an implicit choice construction in the BP model obtained as a result of analyzing a logbook of events. Revealing such a construction occurs when in the analyzed model fragment there is such an intermediate sequence of one or more procedures that an input of the given sequence simultaneously with an output of the final procedure of a fragment is defined by an output of an outside subprocess. Besides, the final fragment procedure has a second input, then there is an implicit relation between the initial and final procedures in the second input.

Type 2 implicit relation: prove the presence of the initial relation on the basis of an simplified scheme of interacting an implicit choice construction with other BP fragments on the base of a relation in input in Pn. The simplification consists in the fact that the sequence of procedures <P2…Pn> is replaced by the single procedure Pn, and the outside subprocess is replaced with the separate procedure P2. The given simplification does not effect on the heart of the proof, as representation of a part of the process as a sequence of procedures or as a common generalized procedure,

realizing the whole subprocess, depends upon the degree of the model details worked out(Fig. 9)

Proceeding from the interactions represented in Fig.2, formulate a set of conditions defining the implicit relation of the given type:

- There is no explicit relation between initial P1 and final Pn procedures of the fragment under study.
- The initial procedure of fragment P1 in the model, obtained on the basis of analyzing the logbook of operations, has two outputs (or more – in the general case) that are inputs:
    - for the initial procedure $P_j, j = \overline{2, n-1}$ of the outside subprocess
    - for procedure P2 of the current BP fragment.
- There is an implicit relation in the sense of expression between procedures P1 and Pn.
- The results of implementing procedure Pn-1 are used as input ones for procedures Pk and Pn
- The final procedure of the outer subprocessPk is not connected by an indirect relation with initial procedure Pn.

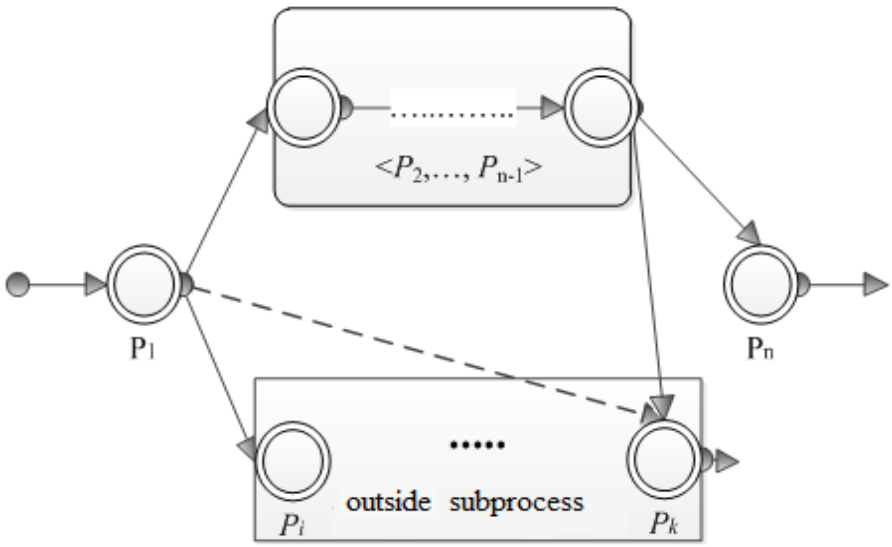


Fig.9. Implicit relation of type 2: outputs of analyzed fragments - inputs of an outside subprocess.

To be more exact, such a relation is not guaranteed in the general case. There is a type 2 implicit relation between procedures P1 and Pk when meeting the considered five conditions.

The set of the conditions formulated allows to identify an implicit choice construction in a BP model, obtained as a result of analyzing an event recording logbook, in the case if in the current model fragment there are two (in the general case more than two) variants of a process following that terminate with procedures Pk and Pn. Their implementation depends in which chain - <P2 …Pn> or on the outside subprocess <Pi ..Pk> the realization of a BP will follow after implementing the initial procedure. Then, if under the real execution of a process, reflected in an event recording logbook, procedure Pn has been implemented, it means that the chain <P2…Pn-1> has been realized.

Type 3 implicit relation: the outside subprocess is implemented concurrently affecting the sequence <P2…Pn-1> (Fig. 10).

On the basis of analyzing the presented scheme formulate a set of conditions defining the relations of the given type:

- Output of initial procedure P1 – input into the intermediate fragment <P2…Pn>;
- Output of the initial procedure of outside subprocess Pi;
- Input into the intermediate fragment <P2…Pn>;
- Output of the procedure Pn-1 - input into the final procedure of outside subprocess Pk;

Output of the last procedure of the intermediate sequence of the current fragment Pn-1 input into the final procedure of outside subprocess Pk.

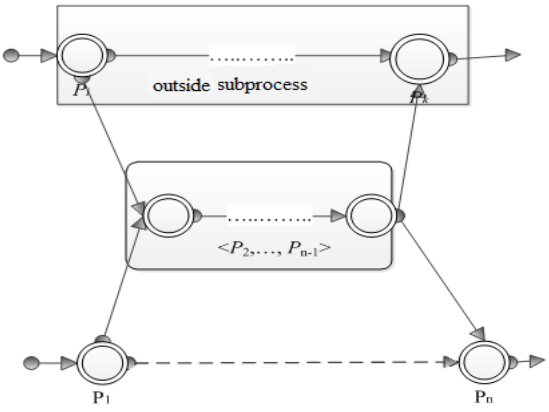


Fig.10. Type 3 of implicit relation is implemented concurrently

On the basis of analyzing the presented scheme let us formulate a set of conditions defining the relations of the given type:

- There is no explicit relation between initial P1 and final Pn procedures of the fragment under study.
- There is an indirect relation between initial P1 and final Pn procedures of the fragment under investigation;
- The first procedure P2 of the intermediate sequence <P2 … Pn> has two inputs:
    - from initial procedure P1 of the current fragment of the BP of an outside subprocess.
    - from initial procedure Pi of an outside subprocess.
- The results of implementing procedure Pn-1 are used as input ones for procedures Pk and Pn-1.
- The final procedure Pk of the outside subprocess is not indirectly related to initial procedure P1. To be more exact, such a relation is not guaranteed in the general case.

There is a type 3 implicit relation between procedures Pi and Pk when satisfying the considered five conditions.

The conditions, given above, allow to identify a type 3 implicit choice construction in that case if an outside subprocess is implemented concurrently with the current BP fragment, which results in the appearance of a construction with two input procedures and two output ones, defining two variants of a process following. Implementing this in that variant depends upon which chain - <P2 … Pn-1> or on outside subprocess <Pi … Pk> the BP will be realized after executing initial procedures. Then, if implementation of a procedure is represented in the event recording logbook, it means that execution of the chain <P2 … Pn-1> has occurred and, hence, there is an implicit relation between the procedures.

Type 4 implicit relation: the outside subprocess is implemented concurrently and with depending upon the sequence <P2 … Pn-1>.

The given relation is the detail of interactions shown in Fig. 2 and based upon the concurrent implementation of the fragment under consideration and the outside subprocess; the outside subprocess start is defined by a current fragment, and its completion effects the implementation of the current fragment (Fig. 11):

- Output of procedure P2– input into the initial procedure of outside subprocess Pi;
- Output of the final procedure of outside subprocess Pk - input into the final procedure of the intermediate sequence of the current fragment Pn-1.

On the basis of analyzing the scheme, shown above, formulate a set of conditions allowing to formalize the relations of the given type:

- There is no explicit relation between initial P1 and final Pn procedures of the fragment under study.
- There is an indirect relation through the sequence <P2 … Pn-1> between initial P1and final Pn procedures of the fragment under investigation.
- The first procedure P2 of the intermediate sequence has two outputs:
  - into the subsequent procedure of the current BP fragment;
  - into the initial procedure Pi of an outside subprocess.

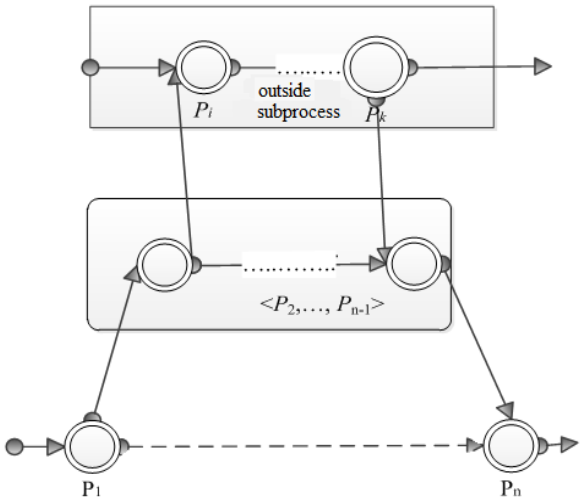


Fig.11. Type 4 of implicit relation: outside subprocess is implemented concurrently

- The results of implementing an outside subprocess procedure are used as input ones for procedure Pk;
- The final procedure of the outside subprocess is not indirectly connected with the initial procedure. To be more exact, such a relation is not guaranteed in the general case.

There is a type 4 implicit relation between procedures Pi and Pk when meeting the considered five conditions.

Predicate models of implicit relations between procedures which are obtained in the given subsection, represent different variants of concurrent and sequential execution of the current fragment of a BP, presenting the current situation, with other subprocesses.

The mentioned models allow formally representing implicit knowledge about interactions between procedures and, therefore, increasing the adequacy of the model obtained as a result of analyzing of a logbook of recording BP events.

Conclusion

The obtained algebraic logical models of implicit choice constructions represent logical nets in the form of a system of predicates representing interactions between the procedures of the construction under study. Such models allow to present implicit cause-and-effect relations between appropriate procedures which are not directly presented in a logbook of recording BP events.

Basing on examples of operation of constructed logical nets, one can see that only one predicate is calculated in the bigger part of steps, which states that these logical nets operate practically in a sequential regime and have no advantages over models of the same processes in the form of multi-place predicates. However, if one takes into account that only small key fragments of real BPs have been covered in the examples considered, and full models often contain a lot of procedures which can be implemented simultaneously, then the advantages of the models in the form of systems of binary equations become evident.

The practical aspect of the results obtained consists in the following. Implementing a logical net, realizing implicit choice constructions, ensures a possibility for obtaining a logbook of recording events with representation of implicit interactions between procedures which creates conditions for developing methods of revealing implicit choice constructions.

Bibliography

[Agrawal, 1998] R. Agrawal, D. Gunopulos, and F. Leymann.Mining Process Models from Workflow Logs. [Text]/ In Sixth International Conference on Extending Database Technology – 1998 – 469-483 pages.

[Aalst, 2003] W.M.P. van der Aalst, B.F. van Dongen, J. Herbst, L. Maruster, G. Schimm, and A.J.M.M. Weijters. [Text]/ Workflow Mining: A Survey of Issues and Approaches. Data and Knowledge Engineering, 47(2):237–2003 – 267 pages.

[Medeiros, 2003] A.K.A. de Medeiros, W.M.P. van der Aalst, and A.J.M.M. Weijters. Workflow Mining: Current Status and Future Directions. In R. Meersman, Z. Tari, and D.C. Schmidt, editors/ [Text]/ On The Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE, volume 2888 of Lecture Notes in Computer Science, pages 389 -406. Springer-Verlag, Berlin, 2003.

[Aalst, 2004] W.M.P. van der Aalst, A.J.M.M. Weijters, and L. Maruster. Workflow Mining: Discovering Process Models from Event Logs. [Text]/ IEEE Transactions on Knowledge and Data Engineering –2004– 16(9):1128-1142 pages.

[Desel, 2005] J. Desel and J. Esparza. Free Choice Petri Nets [Text] / volume 40 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge, UK, 1995.- 256 p.

[Li, 2003] M. Z. J.Q. Li, Y.S. Fan. Timing constraint workflow nets for workflow analysis. [Text] / IEEE Transactions on Systems, Man, and Cybernetics, part A: Systems and Humans, 33(2):179–193, March 2003.

Authors' Information

Olga Kalynychenko *- PhD, associate professor of Software Engineering department, Kharkov National University of Radio Electronics Ukraine; Lenin av., 14, 61166 Kharkov, Ukraine; e-mail: okalinichenko@mail.ru*

Vira Golian *- PhD, Software Engineering department, Kharkov National University of Radio Electronics Ukraine; Lenin av., 14, 61166 Kharkov, Ukraine; e-mail: veragolyan@yandex.ru*

Nataliia Golian *– PhD student of Software Engineering department, Kharkov National University of Radio Electronics Ukraine; Lenin av., 14, 61166 Kharkov, Ukraine; e-mail: veragolyan@yandex.ru*

INTELLIGENT ANALYSIS OF MARKETING DATA

Łukasz Paśko, Galina Setlak

***Abstract**: The main goal of this paper is to present and evaluate the possibility of using the methods and tools of Artificial Intelligence and Data Mining to analyze marketing data needed to support decision-making in the process of market segmentation. This paper describes the application of Kohonen's Neural Networks and Classification Trees (including tools such as CART-Classification and Regression Tree, Chi-squared Automatic Interaction Detector (CHAID) and Boosted Tree) to solving problems of classification and grouping of data. The main part presents the results of market segmentation that can be used by the company producing household products. Finally conclusions and further research plans have been described.*

***Keywords**: data analysis, artificial intelligence, data mining, classification, clustering, Kohonen's neural networks.*

***ACM Classification Keywords**: I.2.m Miscellaneous : I.2.6 [Artificial Intelligence]: Learning – Connectionism and neural nets; I.5.1: Models – Neural nets; I.5.3: Clustering – Algorithms.*

Introduction

In order to effectively run a modern enterprise operating on international markets, in conditions of intensive and constantly growing competition, there is a need for knowledge that is often hidden in massive data sets. Every organization conducts its business with the use of huge amount of information. This information constitutes the sources of knowledge regarding firm's processes and functions, and is essential for making informed decisions.

Today the words of Philipp Kotler, well-known specialist in the field of marketing become more and more important. As he says: "If the data cannot be transformed into information, which will be the basis of knowledge, and knowledge - the source of wisdom, we lose far more than we get." [Kotler Ph., 1984]. These words show the importance and the need to develop modern tools and systems for the processing and analysis of all these huge streams of information in order to obtain the knowledge necessary to make optimal decisions in the process of firm's management.

The main responsibility of marketing analysis is to provide managers with very important market knowledge which concerns the potential markets, specificity of demand and the trends related to pricing as well as advertising effectiveness. It gives a manager an opportunity to learn how the characteristics of the market, distribution channels and strategies to stimulate sales look like.

Traditionally, marketing data analysis uses statistical methods such as regression and correlation analysis or discriminant and factor analysis. As noted by Stanimir "The effectiveness of marketing activities may be multiplied by interpreting customer behavior using modern IT solutions." [Stanimir, 2006]. This is why nowadays marketing research is supported by advanced tools, methods and techniques of artificial intelligence.

In recent years, the field of research known as "data mining" has been developing very intensively. Data mining is primarily focused on automating the analysis of large data sets stored in databases and/or data warehouses. Data mining is about finding hidden patterns or relationships in large data sets. The knowledge regarding these discovered patterns may be used as a basis for decision making processes. Data mining is also known as an intelligent analysis of the data, because the paradigm is based on the use of artificial intelligence methods and techniques (neural networks, genetic algorithms) [Hand David et all, 2005], [Larose , 2006], [Cios K., 1998].

In this paper the evaluation of the possibility of using the methods and tools of Artificial Intelligence and Data Mining to analyze marketing data, needed to support decision-making in the process of market segmentation has been presented. The paper describes the study of the application of Kohonen's Neural Networks and Classification Trees (including CART-Classification and Regression Tree, Chi-squared Automatic Interaction Detector (CHAID) and Boosted Tree) to solving problems of classification and grouping of marketing data. The main part presents the results of market segmentation that can be used by the company producing household products. Finally conclusions and further research plans have been described.

Theoretical foundations

Classification and grouping belong to basic tasks of the data mining. Problems of classification are very common (which products should be offered on which markets, which customers are credible and which are not, recognition of signals and images etc.) and are being solved in the everyday business activity, in the process of managing organization, in the medical or technical diagnostics, and in data analyses as well as measuring experiments.

Especially significant are the developments of tools and systems that are able to effectively support the problem solving in situations of high complexity, large number of problem's parameters or time constraints imposed on the solution. Problems of classification and grouping of marketing data belong to the above mentioned categories and usually require sophisticated computational techniques and methods [Migut, 2010], [Mynarski, 2010] , [StatSoft, 2010].

Classification is about assigning the object to one of the model classes based on selected distinctive features. The process may be called classification when the categories to which elements of input set will be assigned are determined in advance. The term classification is also defined as an automatic determination of objects affiliation to specified class on the basis of their images [[Adamczak, 2001], [Hand David et all, 2005], [Stąpor, 2011], [Szczuka, 2000]]. Classification is the most frequently solved problem both in technology and economy. Formally, classification is defined as the process of mapping data into the set of predefined classes.

$$f_c : R^p \supset X \rightarrow C, \tag{1}$$

where $C = \{C_1, C_2, ..., C_n\}$ is the finite set of classes, whereas the set $X \subset R^p$ is the attribute space, and the decision about the classification result is based on these attributes. Classification mapping f_c divides space X into n decision areas, grouping the attributes patterns that belong to one category [Zieliński, 2000]. Input data is the set of examples, observations and samples which are the list of the descriptive features values. Output data constitutes a model (classifier). The main aim of data classification is to build formal classifier. The classification process consists of two stages: building a model and testing unknown values.

Grouping also known as clustering is a primal in relation to classification because is responsible for defining classes. The process may be called grouping when the number of groups and the ranges of distinctive features values are not known before the division of the input set. Grouping is about matching similar and separating different objects [Hand David et al., 2005], [Li S., 2000], [Żurada et al., 1996]. It also means the division of set of elements into subsets on the basis of distinctive features detected during the process of division.

Neural networks are widely used as classifiers [Jang et al., 1997], [Moon et al., 1998], [Takagi et al., 2000], [Setlak G., 2004]. Classification and clustering problems have been addressed in many research works and disciplines such as statistics, machine learning as well as databases. The basic algorithms of the classification methods are presented in [Nauck et al., 1997], [Stąpor, 2005], [Żurada et al., 1996].

The applications of the clustering procedure can be divided into following categories [Hand David et all, 2005]:

- hierarchical form trees in which the leaves represent particular objects, and the nodes represent their groups. The higher level concentrations include the lower level concentrations. In terms of hierarchical methods, depending on the technique of creating hierarchy classes (agglomerative methods and divisive methods);
- graph-theoretic clustering,
- fuzzy clustering,
- methods based on evolutionary methods,
- methods based on artificial neural networks.

The most often used classification methods, also applied to marketing data analysis, are the following: decision trees (aka classification trees) [Li S., 2000], [Mynarski, 2010], methods of naive Bayes classification [Migut, 2010], [Stanimir et al., 2006], memory classification (e.g. nearest neighbor method or neural networks [Li S., 2000], [Szczuka, 2000]). Nowadays for solving problems of classification and grouping scientists are using also artificial neural networks [Hand David et al., 2005], [Larose, 2006], [Setlak, 2000], [Stąpor, 2011] and algorithms of fuzzy grouping (*aka* fuzzy clustering algorithms) [Czogala et al., 2000], [Jang, 1997]. What is more, the solutions that are combination of several artificial intelligence tools (neural networks, fuzzy logic, genetic algorithms) are applied, what enables to create more efficient hybrid neuro-fuzzy classifiers in which genetic algorithms are used for a teaching process [Nauck, 1997], [Rutkowska, 2000], [Setlak, 2001], [Setlak, 2008], [Lotfi Zadeh, Rutkowska, 2000].

In this research, artificial neural networks and decision tree algorithms have been used as a basic tool for creating classifiers. Neural Networks have been selected because they possess such characteristics as approximation abilities, interference immunity and adaptability. However, it is worth to remember that it may be good idea to use neural networks along with other AI tools such as decision trees, expert systems, and fuzzy sets as well as fuzzy logic. Such combination enables to design more efficient support systems, free from neural networks deficiencies.

Experiment

An experiment includes several tasks of vacuum cleaners market segmentation. Market segmentation is about dividing market into smaller parts called segments. Segments are created according to segmentation criteria. These criteria may be related to customers'

preferences, method of purchase, gender, age or other characteristics selected by marketer.

After segmentation is done, company has to choose the segment that is best suited for specific product. The primary goal of segmentation is customer preferences analysis. The secondary is the product positioning, which is the process of giving the product specific features that will differentiate it from similar products offered by competitors [Migut, 2010], [Stanimir et al., 2006].

Before the segmentation is performed the criteria of segmentation have to be determined. Next important step is to identify, if it is possible, optimal number of segments (groups or classes). In specialist literature two types of segmentation are distinguished: descriptive segmentation and predictive segmentation.

Descriptive segmentation is usually used in situations where:

- there is lack of criteria used for groups selection,
- all the variables (attributes describing objects) are independent variables,
- there is a lack of information allowing the use of methods of learning with the teacher, and one can use only the methods of so-called not-directed data mining (learning without a teacher) [Migut, 2010].

Descriptive market segmentation may be conducted with the use of such methods and tools as agglomeration method, the method of K-means and one of the modern methods – Kohonen's Neural Networks.

Table 1. Characteristics of the products (vacuum cleaners)

The attribute	Description	The data type
ENGINE_W	Engine power, W	Numerical
PRICE	Price	Numerical
FILTR_SYS	Advanced filtration system	{yes, no}
AUTOFUNC	Automatic power control	{yes, no}
AUTOCORD	Auto cord rewinder	{yes, no}
SPD_CTRL	Electronic adjustment of suction strength	{yes, no}
NOISSYS	Noise suppression system	{yes, no}
WASH	Wet cleaning option	{yes, no}
VIEW	Style and design	{yes, no}
FEATURE	Additional features	{yes, no}
BRAND	Brand	{yes, no}
SERVICE	After sales service	{low, medium, high}

Predictive Segmentation is used in situations where:

- it is possible to determine the criterion of segmentation,
- the variables are independent,
- directed data mining methods based on learning with the teacher may be used.

The data sets used in the study have been developed as a result of marketing research (surveys) conducted for company producing vacuum cleaners in years 2003-2005. The data includes vacuum cleaners characteristics. For these products, according to research goals, the tasks of classification and grouping have been carried out. Table 1 contains the input parameters (products' characteristics) for the classification and grouping processes.

An output parameter, in a prepared data set, is one of the market segments that is selected in the process of classification. It is marked in the training set with CLASS label.

Descriptive segmentation with Kohonen's Neural Networks

At first, the process of data clustering has been done in Statistica Neural Networks environment with the use of Kohonen's Neural Network.

Kohonen's Neural Network is also called the self-organizing maps (SOM) due to the way of learning – it uses unsupervised learning approach. Neural Networks developed by Teuvo Kohonen in 1982 [Kohonen,1989], [Kohonen,1990] constitute the special category of Artificial Neural Networks. These are unidirectional networks, made up of two layers. In the first layer (input layer) are only neurons corresponding to the input signals for transmission from the data source to the network. Each neuron of the input layer is connected to all output neurons (so called full network). Output layer acts as both computing and presenting the results. Neurons form a topological map, thanks to which, data clusters found in the data set may be observed.

The basic method of self-organizing networks learning, including Kohonen's network, is a competitive learning method [Kohonen,1989]. The method is based on competitive learning in which only one output neuron in the group is active at a given moment. All output units compete with one another, so this rule is sometimes called the "winner takes all". Purpose of this type of network is grouping or classification of input patterns. It is done in accordance with the principle that similar input signals trigger the same output units of the neural network. Groups are defined based on the correlation of the input data. Kohonen's network is able to process complex input signals and thanks to this it may be used to test a set of products that may have both quantitative and qualitative attributes. It was decided that during the learning process, the CLASS attribute is not taken into account. It will be treated as an attribute, which the groups found by the network will be

compared with. Disabling CLASS variable in the process of learning is necessary to ensure that the comparative analysis of clusters found by the network and market segments gathered in CLASS attribute is reliable. Besides CLASS attribute, all other variables have been selected for analysis.

The data that will be used by the neural network has to be properly prepared in advance. All the records in the data set should contain all values of attributes. In addition, all the values should be stored in the same structure for all records. The data set has met these assumptions. Therefore, the initial stage of data processing (called preprocessing) is limited to encoding the attributes' values. In the case of continuous attributes (ENGINE_W and PRICE) their maximum and minimum values in the data set are known. In addition, it is assumed that the network learned will be used to group objects which attributes' values do not exceed the scope defined in the training set. For encoding continuous attributes min-max normalization method was used. It allows to normalize the value of the variable that will belong to <0,1> interval. It is done according to the formula (2):

$$X^* = \frac{X - \min(X)}{range(X)} \tag{2}$$

where:

X* - Scaled continuous value attribute of the record,

X - The attribute value of the record before scaling,

min(x) - Minimum value of the attribute in the data set,

range(x)=max(x)-min(x) - The range of attribute values in the entire data set.

In the case of SERVICE attribute, neural network uses three input neurons corresponding to one of three possible attribute states: "service = low", " service = medium ", "service = high". The possible tags' values are true (1) or false (0). All other attributes are encoded using two values: 0 and 1. In the next step, the data set used for the analysis has been divided into three subsets: for learning, for validation and for testing. Training set is responsible for the proper modification of neuronal weights during learning process. A validation set controls learning error value, what allows the selection of the best network training algorithm, and stops the process if the symptoms of network over-training have been identified.

The validation set is a basis for an independent test to check the correctness of the network operation and will be used at the end of the learning process. For the validation and test sets, 39 cases have randomly been selected. The remaining 116 records constitute training set. Table 2 shows the number of cases of each segment divided into training, validation and testing sets.

Table 2. Number of cases in the collection used in Statistica Neural Networks

	CLASS attribute value (market segments)				Sum
Set	**m1**	**m2**	**m3**	**m4**	
Training	25	47	27	17	116
Validation	8	11	10	10	39
Testing	13	14	5	7	39
Sum	46	72	42	34	194

The assumption related to data division into training, validation, and testing sets was to keep similar proportion of training cases to total number of cases for every value of CLASS attribute. Thanks to drawing process that has been used for data set division, this assumption has been met. The result is that the training set is as representative as the entire data set. Such prepared data set was analyzed with the use of Kohonen's network. The structure of the network is shown in Figure 1. It includes one input neuron for each quantitative and binary attributes and three neurons for SERVICE attribute (one neuron for each possible value of this attribute). In the output layer there are 98 neurons arranged according to 14 × 7 configuration.

The size of the output layer has been determined experimentally. The minimum number of neurons needed to create a topological map that reflects the market segments is four (each neuron corresponds to a different segment). However, after learning the network with 2×2 output layer and analysis of its behavior it turned out that the error it makes is not acceptable. Every neuron was the winner for the cases related to different market segments. Therefore it was not possible to tag every neuron with market segment label. Extending the size of topological map enabled the identification and tagging of clusters recognized by the network.

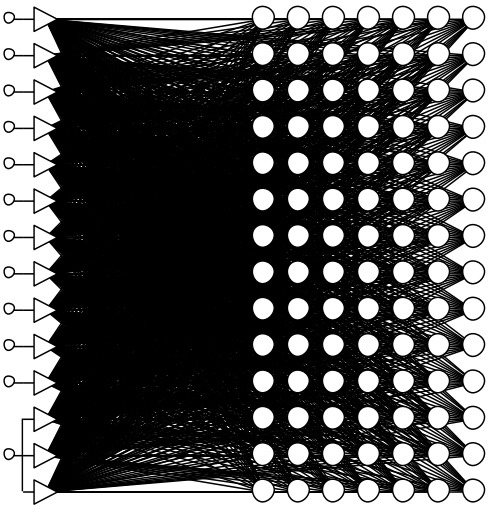


Figure 1. Architecture of Kohonen's network developed for the research

Teaching the Kohonen's network in Statistica Neural Networks environment was conducted with the Kohonen's algorithm. Learning consists of three stages. In the first one (called preliminary learning) network "is reading" the data set and initially arranges clusters on the topological map. The second stage is called tutoring or fine tuning, and provides a better fit of the output layer neurons to the training set by appropriate adjustment of the position of individual neurons. For both phases the values of three learning parameters have been experimentally determined (Table 3).

Table 3. The parameter values Kohonen network learning

	Number of epochs (periods)	**Learning rate**		**Neighborhood**	
		Initial value	**The final value**	**Initial value**	**The final value**
Initial learning	200	0.50	0.10	7	6
Tutoring	500	0.10	0.01	6	0

Number of epochs is the number of iterations of the learning algorithm. During a single iteration the network examines the training set and modifies the weights. It was assumed that in the first two stages of learning there is a need for 200 periods. In the second step, for more precise clusters identification, this number was increased to 500. Adoption of

smaller number of epochs resulted in deterioration in cluster recognition done by network. However, the greater number of iterations resulted in only a slight reduction of the error.

Learning rate is responsible for a speed of learning. The higher this value is, the stronger the network weights are modified after every iteration of the learning algorithm. It was determined that during the initial learning this factor had the greatest value. Over the iterations, the learning rate was decreasing linearly. During the process of network training the value of learning rate adopted was five times smaller than during the first stage of learning. This allows the Kohonen's network to learn more slowly but more accurately.

Neighborhood size determines the number of neurons within the winning neuron, which are involved in the adaptation process. After selecting the winner from the training set for a particular case, the weights of the winner and all neurons within the neighborhood have been modified. It causes that these neurons have a greater chance of winning the competition in the future, when similar case occurs in the training set. In the first stage of learning greater value of neighborhood has been adopted in order to enable the creation of clusters on the topological map. In the second stage of learning the neighborhood size was decreased to 0. As a result only weights of winners were modified, what has made the clusters boundaries more clear.

After learning process the Kohonen's network has been run for all cases from the data set. A large number of wins of the neuron indicates the existence of the center of cluster in the given place of the topological map. All the neurons of the output layer, which won at least once, have been named. For labeling the neurons the CLASS attribute values were used (label "mn" denotes winning neuron for the cases in the segment tagged "n"). The results of labeling process are shown in Figure 2.

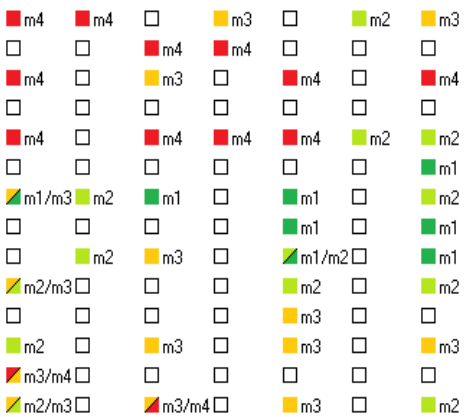


Figure 2. The topological map of Kohonen's network with segments' labels next to winning neurons

On the topological map presented in figure 2 it is possible to observe the regions, which include neurons that won for the cases from one specific market segment. However, in some areas of the map it is quite difficult to clearly determine the boundaries between clusters, because some neurons that are next to each other and have labels of different segments. What is more, some neurons won for cases belonging to two different segments of the market. Therefore, in the next step the cases that belong to hardly recognizable areas of the map have been analyzed.

During the comparative analysis of maps regions it turned out that there are groups of products with similar characteristics, yet belonging to different segments of the market. This is the reason why the map has been labeled again. The new labels reflect the boundaries of clusters identified by the Kohonen's network and now take into account the results of the comparative analysis in the areas that have initially been difficult to interpret. A topological map after re-labeling is shown in Figure 3.

In order to distinguish neurons after re-labeling, they have been marked with "cn" labels, where "n" is a number of recognized market segment. Then, based on new topological map for each case from data set new market segment has been assigned. Information about which product belongs to which market segment is stored in the data set as the CLUSTER attribute's value. This attribute is the second dependent variable (apart from CLASS variable). All analyzes have been conducted with the use of decision trees for both dependent variables. It enabled to assess what was the impact of Kohonen's network on decision trees built.

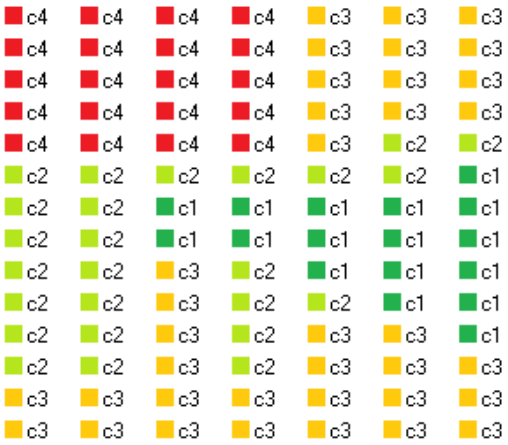


Figure 3. Kohonen's network topological map after re-labeling neurons

Classification and Regression Trees

Statistica Data Miner contains a wide range of techniques for classification as well as construction and implementation of appropriate models. For classification task one can use such techniques as: Classification Trees Models, General Classification and Regression Trees (GTrees), general CHAID models, cluster analysis, cluster analysis using generalized EM and k-means (with a cross test). In the study presented in the paper, the first models have been developed with Classification and Regression Trees (CART).

The main idea behind the classification and regression trees is to present a set of data in the form of a decision tree. Algorithms that allow to generate decision trees belong to the teacher based learning algorithms category (also known as supervised learning). In these algorithms the training set with the qualitative goal variable that divides cases from the training set into classes has to be provided. The algorithm learns what values of attributes correspond to each class of goal variable. Generated tree can classify not only cases from the training set, but it should also have the ability to generalize the knowledge gained. This allows to classify new cases, that are not present in the training data. Thus it can be a part of a decision support system.

The decision tree is a directed graph consisting of decision nodes and leaves, interconnected by branches. Each decision node corresponds to an attribute of the data set. The branches coming out from the node symbolize the possible values of a given attribute. Each branch may lead to the next decision node or a leaf node. Decision nodes divide the data set into subsets. The leaves contain a solution of the classification problem, which assigns a specific class to the subsets of cases. In the decision nodes there are variables that are able to divide the data set according to the goal variable.

Classification and Regression Tree (CART) is an algorithm that can generate a strictly binary tree, where each node can divide the data set into two subsets only. Trees of this kind are rarely used in solving marketing problems. Nevertheless it was decided to compare the predictive abilities of the CART trees created for CLASS and CLUSTER dependent variables. For every variable the distinct trees with the similar level of complexity (with regard to number of decision nodes and leaves) have been selected for analysis.

Program generated 8 CART decision trees with different degree of pruning for both CLASS and CLUSTER dependent variables. The pruning of decision tree protects it from overfitting, the situation in which tree depends too much on irrelevant features of the training instances, with the result that it performs well on the training data but relatively

poorly on unseen instances. CART algorithm is able to properly trim the nodes and branches of the tree. The resulted tree makes bigger classification error but it is less complicated. Lower tree complexity results in the ability to knowledge generalization, what allows for better classification of unknown cases during learning. In order to select the best pruned tree V-fold cross-validation was used. It is designed especially for small data sets, because it does not require a separate set of test cases isolated from the training set. It draws the V samples from the training set, which will serve as test sets. Then they are tested with varying degrees on trees with different degree of pruned branches. The result is the cost of cross-validation calculated on the basis of the results of classification of each of the trees. As best pruned tree is selected the one that is the least complex and at the same time its cost of cross-validation is the smallest one. The program has selected the models for CLASS and CLUSTER variables with regard to the selection criteria described above. Table 4 presents the results.

Histograms placed in the leaves nodes represent the distribution of the dependent variable for training cases assigned to the leaf. In case of tree for the dependent variable CLASS, the highest classification uncertainty may cause the leaves with numbers 18, 26, 30, 31. Their histograms show that with such complexity of the decision tree, several cases are classified into other segments of the market than they have nominally been assigned. It was decided to analyze these training cases. They have been compared with the cases put by the Kohonen's network in the areas of the topological map that were difficult to describe. It turned out that several records assigned to four of these leaves also belong to these regions of a topological map of the network. This example shows that both the Kohonen's network and the CART algorithm have found a group of products that are difficult to unambiguously classify to one of the four segments described in CLASS variable.

Histograms placed on the tree for the CLUSTER dependent variable show that cases have been classified with higher level of certainty. None of them indicates that to one leaf similar number of cases belonging to different classes has been assigned. Although the CART algorithm used the same predictors during the creating of both two trees, variables that were included in the decision nodes are different. PRICE is the attribute that differentiates CLASS variable the most. For the CLUSTER variable there is SPD_CTRL attribute located at the root of the tree. This also reflects the different approach to the classification of products for both trees.

Comparing the performance of CART trees presented in Table 4 it can be concluded that a better ability to predict has a tree created for CLUSTER variable. It is backed by cross-validation cost which is twice lower than the cost of CLASS variable. In addition, the model for the CLUSTER variable is simpler. The CART algorithm found more easily

the interdependences between attributes when analyzing a set of data for the CLUSTER dependent variable. In the case of CLASS variable, the algorithm needed more attributes to map the data set, and despite of this the tree makes larger predictive error.

Table 4. Summary of selected parameters CART trees

Dependent variable	Number of leaves	Number of nodes making	The cost of cross-validation
CLASS	12	11	0,288660
CLUSTER	8	7	0,113402

Two trees have been generated with the use of the same algorithm, where each of them describes the same data sets. Only difference is in the dependent variable values. Despite the fact that the dependent variable of the second CLUSTER was created on the basis of CLASS dependent variable, these two trees vary considerably with regard to nodes arrangement as well as the number of erroneous predictions.

Chi-Squared Automatic Interaction Detector

CHi-squared Automatic Interaction Detector (CHAID) is one of the oldest algorithms to create decision trees [Kass, 1980]. CHAID tree can be used for classification and regression tasks. CHAID algorithm is very efficient for large data sets. It creates non-binary trees, that is those in which from decision node can go more than two branches. This is the primary feature that enables to distinguish CHAID trees from CART trees.

To select the variables that are included in decision nodes, the algorithm uses the chi-square test. Important parameters that should be defined in the construction of tree are stop conditions. The modification of these parameters affects the degree of pruning the tree. These parameters include such values as: the minimum number of training cases in the node, which is subject to division, the probability of splitting and merging categories and the maximum number of tree nodes. It is assumed that the minimum number of cases for the divided node will be 19. Other parameters have been determined experimentally. The aim was to generate the tree which has the size similar to the size of CART tree previously created for CLASS dependent variable.

It was determined that the probabilities of splitting and merging categories are the same and equal 0.01. These values allowed to generate trees for the CLASS and CLUSTER variables.

When analyzing the CHAID tree for the CLASS variable it can be seen that the histograms placed in the leaves of the tree reveal the relationship, which has already occurred in the

CART tree for this variable. Some leaves allocate cases belonging to different market segments to the same class. This trend has been preserved despite the fact that both the trees are different with regard to nodes arrangement and the variables selection. The root of CHAID tree is the VIEW attribute. CHAID algorithm used to build the tree such variables as BRAND and SPD_CTRL. These variables were not present on the CART tree. Comparing CART and CHAID trees for the CLUSTER variable few differences may be observed. There are the sizes of the trees, the ways the attributes have been selected in the nodes and the presence of leaves that connect cases belonging to different classes. These leaves cause greater prediction uncertainty, already recorded in the CLASS variable. Analyzing the number of cases attributed to the wrong class, it can be said that the CART tree better adapted to the training data. A common feature of two trees is their binary nature, because the CHAID algorithm has not created decision nodes that divide the data set into more than two parts. In case of CHAID trees V-fold cross validation provides the value which is known as the risk of estimation. This measure substitutes the cost of cross validation which is used for CART trees validation. The basic parameters of both trees are presented in table 5.

Table 5. Summary of selected parameters of CHAID trees

Dependent variable	Number of leaves	Number of nodes making	Risk of estimation for cross-validation
CLASS	14	10	0,243523
CLUSTER	11	12	0,144330

Boosted Trees

Boosted Trees are one of the newest methods of data mining. They can be used for quantitative and qualitative variables for regression as well as classification. In this method, the data model is created using the basic decision tree. Although each tree when considered separately gives a large classification error, together they form a model that has an excellent ability to predict. All trees are usually binary trees, and therefore each decision node divides the data set into two subsets. Algorithm is trying to determine the best distribution of data (to create a single binary tree) and calculates the residues for each division, the deviations observation from the mean values. Then another tree is built, which fits the calculated residuals and makes another division of the data set. With every new tree added the error related to a whole sequence of trees is reduced.

Boosted Tree algorithm is able to recognize even very sophisticated relationships among variables and thanks to this is the data fit is perfect. As with other methods, it is adverse

phenomenon, which reduces the prediction correctness of the model. Boosted Tree algorithm solves this problem by creating for each iteration two sets: training and testing. The cases belonging to both sets are randomly selected from the data set. Training cases are used to generate another tree that is added to the trees sequence and will be used for the prediction of residuals calculated for preceding trees in the sequence. Creating trees for subsequent samplings results in the decrease of prediction error and is called stochastic gradient reinforcement. Test cases are not used for creating the tree. They are applied to the process of model verification in a given iteration.

When the model was created two parameters used by the Boosted Tree algorithm were modified: learning factor (responsible for the speed and accuracy of learning) and the size of the test set (the proportion of the number of test cases to the number of all cases in the data set). Several models have been created which differ with regard to the number of trees and the prediction error. In order to allow comparison of models for the CLUSTER and CLASS variables and adopted the same learning parameters were assumed for both models. Experimentally determined value of the learning factor is 0.1 and the size of the test set is 0.3.

While generating both models graphs were created. Figure 4 presents the graph for the CLASS variable and figure 5 - for the CLUSTER variable. In both cases, it can be observed that every new tree added to the model reduces the training set error. However, there is a point in which the test set error is not decreasing any more. When a certain number of trees is exceeded there is an increase in the value of the test sample error, what indicates the occurrence of overfitting. The optimal number of trees in the model is the value at which the test set error is minimal.

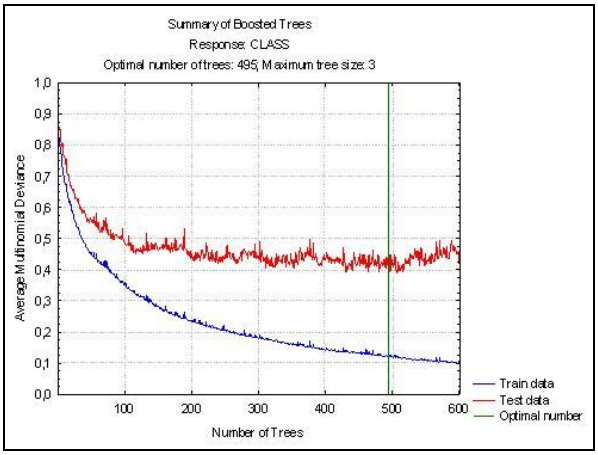


Figure 4. Prediction errors for learning and testing sets in relation to the number of trees in the model for the CLASS dependent variable

It is hard to compare Boosted Tree models with models developed with CART and CHAID algorithms, because their architectures are different. However, the analysis of sequences of trees created generated by Boosted Tree algorithm reveals similarities with the CART models. The first tree in the trees sequence generated for the CLASS dependent variable contains PRICE attribute in the decision node, and the first tree generated for the CLUSTER variable contains SPD_CTRL the attribute. These same attributes are in the CART tree roots for both dependent variables. This means that in case of the analyzed data set both algorithms, Boosted Tree algorithms as well as CART algorithm select variables a similar manner. The parameters of both models generated by Boosted Tree algorithm are summarized in Table 6.

Table 6. Summary of selected parameters of Boosted Trees algorithm

Dependent variable	Learning Ratio	The size of the training set	Number of trees	Risk estimation for cross-validation
CLASS	0,1	0,3	495	0,160714
CLUSTER	0,1	0,3	540	0,035714

As can be seen, the same learning parameters adopted, gave different number of trees and different fit to the data. Risk estimation indicates that the sequence of elementary decision trees for the CLUSTER variable will make the smaller prediction error than the corresponding model for the CLASS variable. This may be confirmed by the graphs created during the construction of models. The application of CLUSTER attribute as a dependent variable resulted in data set that is more "convenient" to learn. The relationships between the variables in this set have been pre-ordered by the Kohonen's network. In case of CART and CHAID decision trees as well as Boosted Tree algorithm early identification of clusters on topological map allowed to easier classification of uncertain records from the data set and therefore reduction of prediction error.

Conclusions and Further Research

For performing analyzes of decision trees, data set describing household products was used. Each case in the data set regards one of the four segments of the market, described with the CLASS dependent variable. Before starting to build decision trees models a grouping of data set with the use of Kohonen's network has been carried out. The aim of the grouping was to identify data clusters corresponding to the market segments. After clusters have been found, all characteristics of products placed by

the network in the areas of topological map that were difficult to identify have been analyzed.

As a result of the analysis the market segments have been assigned again to the products. Information about to which market segment each product belongs has been stored in an additional dependent variable called CLUSTER, which has been included in the analyzed data set. All analyzes were performed for market segments initially defined in the CLASS variable and for market segments described by CLUSTER variable and ordered by Kohonen's network.

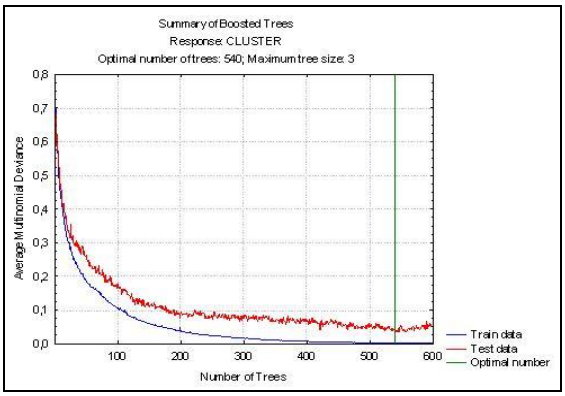


Figure 5. Chart of prediction error learning and test sets relative number of trees in the model for the dependent variable CLUSTER

In conducted analyses, in order to build decision trees, CART, CHAID and Boosted Tree algorithms have been used. These algorithms were used to build predictive models for two dependent variables. Every model developed was assessed with Statistica Data Miner metrics. These metrics include cross-validation cost and risk estimation. The results of the analysis and comparisons of the developed models have been presented in the previous sections. It can be said that all kinds of trees have made the largest predictive error for classifications done for CLASS dependent variable. In case of CLUSTER variable, prediction errors for the test sets (or the V-fold cross-validation) were smaller. This could be due to the fact that the Kohonen's network initially recognized and ordered the interdependences between the attributes that affect the membership of market segments.

The smallest error was made by Boosted Tree algorithm, what confirms that it is the most advanced approach from all the models analyzed. Boosted Tree algorithm adapts very well to the training data, and controls the symptoms of overfitting.

One of the advantages of the Statistica Data Miner environment is that it is possible to quick develop the models with the use of automatic code generator (C / C + +, Statistica

Visual Basic or PMML). After generating the code, all programs may run concurrently in order to determine the value of the dependent variable. This is done thanks to technique called models aggregation. In the case of qualitative dependent variables the voting is used. During the voting process the separate prediction is made by each of the models. As a final result only one class is selected that has been recommended by majority of models. The following tables (table 7 and table 8) show the classification matrix of aggregated model (voting based prediction) as well as decision trees.

Table 7. Classification matrix for models developed for CLASS dependent variable

Observed class	Model	Predicted class			
		m1	m2	m3	m4
m1	CART Tree	28	3	3	
	CHAID Tree	28	5	1	
	Boosted Trees	31	3		
	Voted prediction	28	2	4	
m2	CART Tree	6	32	4	
	CHAID Tree	6	27	9	
	Boosted Trees	1	37	4	
	Voted prediction	4	35	3	
m3	CART Tree		12	58	2
	CHAID Tree	4	7	60	1
	Boosted Trees		3	68	1
	Voted prediction		6	64	2
m4	CART Tree			1	45
	CHAID Tree			1	45
	Boosted Trees				46
	Voted prediction			1	45

After comparing values from both tables it is possible to notice that the aggregated model is trying to average of the prediction errors of individual decision trees. Predictions of this model are worse than the best classifier results (Boosted Tree), but it gives better results than CART and CHAID trees. For several classes the results for aggregated model are the same as those for CART tree. If you want to use an intermediate model and avoid generating aggregated models for data set, CART algorithm should be used. What is more, the aggregated model demonstrates the same tendency as decision trees acting separately do. This model better classifies the cases according to CLUSTER dependent variable. This confirms the positive impact of the Kohonen's network on analyzed data set.

Table 8. Classification Matrix for models developed for the CLUSTER dependent variable

Observed class	Model	Predicted class			
		c1	c2	c3	c4
c1	CART Tree	34		2	
	CHAID Tree	36			
	Boosted Trees	36			
	Voted prediction	36			
c2	CART Tree		37	6	
	CHAID Tree	3	37	3	
	Boosted Trees		43	1	
	Voted prediction		37	6	
c3	CART Tree		3	77	1
	CHAID Tree	3	3	74	1
	Boosted Trees		1	79	1
	Voted prediction		3	77	1
c4	CART Tree		3	1	30
	CHAID Tree			4	30
	Boosted Trees				34
	Voted prediction			1	33

The developed models of decision trees can also be used in combination with neural networks. Neural networks and decision trees are two types of predictive models that differ with regard to many features. The combination of such discrete models is possible thanks to a technique called stacking. It involves the use of at least two models to classify cases from the same data set. Then the results of the classification are passed to the next model, which tries to merge the results and provide a final solution to the problem of prediction.

In the case of developed models it is possible to make separate predictions with CART, CHAID, and Boosted Tree algorithms. The results generated by each of them can be transferred to the neural network input. The network will learn how to combine the results of individual trees so that the final predictive model gave the best possible results. This approach is called "meta-learning", as the collective model learns on the basis of what other models learned before.

Another way of combining decision trees and neural networks may be to use tree-building algorithms to determine what attributes of the data set are the most significant. These attributes can divide the data set in the best possible way and therefore they are placed in the nodes of decision tree. The higher given attribute is located in the tree (closer to the root of the tree), the better the data set is divided. The most significant

variables can be used as inputs to the neural network. After learning process based on such "truncated" data set, network can easily recognize all interdependences between variables. It will result in a better ability to predict new cases, which are not present in the training data set.

Bibliography

[Kotler Ph., 1984] Kotler Ph.: Marketing Essentials, Prentice-Hall International, Englewood Cliffs, 1984, 733 pp.

[Adamczak, 2001] R. Adamczak. Zastosowanie sieci neuronowych do klasyfikacji danych doświadczalnych, Praca doktorska, Uniwersytet M. Kopernika, Toruń, 2001.

[Hand David et al., 2005] Hand David, Mannila Heikki, Smyth Padhraic: Eksploracja danych, WNT, Warszawa, 2005

[Cios et all, 1998] Cios K., Pedrycz W., Świniarski R. Data mining methods for knowledge discovery, Kluwer, Norwell MA, 1998

[Czogala et al., 2000] E.Czogala, Lęski J.: Fuzzy and Neuro-Fuzzy Intelligent Systems, Physica-Verlag, A Springer-Verlag Company, Heidelberg, New York, 2000

[Jang et al., 1997] Jang S.R., Sun C.T., Mizutani E.: Neurofuzzy and Soft Computing, Prentice-Hall, Upper Saddle River 1997, p. 245.

[Kass et al., 1980] Kass, Gordon V.; An Exploratory Technique for Investigating Large Quantities of Categorical Data, Applied Statistics, Vol. 29, No. 2 (1980), pp. 119–127

[Kohonen, 1990] Kohonen T.: Self-organizing Maps, Proc. IEEE, 1990, 78, NR.9, pp. 1464-1480.

[Kohonen,1989] Kohonen T.: Self-organization and associative memory, Berlin, Springer Verlag,1989

[Larose, 2006] D. T. Larose, Odkrywanie wiedzy z danych, Wyd. Nauk. PWN, Warszawa, 2006

[Li S., 2000] Li S.: The Development of a Hybrid Intelligent System for Developing Marketing Strategy, Decision Support Systems, 2000, Vol 27, N4

[Migut, 2010] Migut G.: Zastosowanie technik analizy skupień i drzew decyzyjnych do segmentacji rynku, StatSoft, Materiay Seminarium StatSoft "Zastosowanie nowoczesnej analizy danych w marketingu i badaniach rynku", Kraków, 2010.

[Moon et al., 1998] Moon Y.B., Divers C.K., and H.-J. Kim: AEWS: An Integrated Knowledge-based System with Neural Network for Reliability Prediction // Computers in Industry, 1998, Vol.35, N2, pp.312-344.

[Mynarski, 2010] Mynarski S.: Metody ilościowe i jakościowe badań rynkowych i marketingowych, StatSoft, Kraków, 2010

[Nauck et al.,, 1997] D.Nauck, F. Klawonn, R.Kruse: Foundations of Neuro-Fuzzy Systems, J.Wiley&Sons, Chichester, 1997.

[Rutkowska, 2000] Rutkowska D.: Implication-based neuro-fuzzy architectures.- Applied mathematics and computer science, V.10, N4, 2000, Technical Unieversity Press, Zielona Gora, 675-701.

[Setlak, 2001] Setlak G.: „Fuzzy Neural Networks in Intelligent Manufacturing Systems", // Proceedings of the International Workshop on Intelligent Data Acquisition and Advanced Computing Systems, Copyright Clearence Center IEEE, Piscataway (USA), Ternopil (Ukraine), Foros, 2001, pp.203-206. and http://ieeexplore.ieee.org/servlet/opac?punumber=7498

[Setlak, 2008] Setlak G.: The Fuzzy-Neuro Classifier for Decision Support //International Journal Information Theories and Applications, Pub. of the Institute of Information Theories and Applications FOI ITHEA, Sofia, 2008, Vol. 15, N.1, PP.22-28.

[Setlak, 2004] Setlak G.: Intelligent Decision Support System, // LOGOS, Kiev, 2004, (in Rus.), pp. 250.
[Setlak, 2000] Setlak G.: Neural networks in intelligent decision support systems for management // Journal of Automation and Information Sciences, Kiev, N1, 2000r., pp. 112-119.
[Stąpor, 2011] Stąpor K.: Metody klasyfikacji obiektów w wizji komputerowej, Wydawn. Naukowe PWN, Warszawa, 2011
[Stąpor, 2005] K. Stąpor. Automatyczna klasyfikacja obiektów, Wyd. Exit, Warszawa, 2005.
[Stanimir et al., 2006] Red.: A. Stanimir: Analiza danych marketingowych. Problemy, metody, przykłady, Wydawnictwo Akademii Ekonomicznej we Wrocławiu, 2006
[StatSoft, 2010] Praktyczna analiza danych w marketingu i badaniach rynku, materiały z konferencji StatSoft Polska, wrzesień 2010
[Szczuka, 2000] Szczuka M.: Metody symboliczne i sieci neuronowe w konstrukcji klasyfikatorów, autoreferat rozprawy dokt., Uniwersytet Warszawski, Warszawa, 2000
[Lotfi Zadeh, Rutkowska, 2000] Ed. By Lotfi A.Zadeh and D. Rutkowska: International Journal of Applied Mathematics and Computer Science, Special Issue: Neuro-fuzzy and soft Computing.– Zielona Gora : Technical University Press, Vol. 10, N4, 2000.
[Zieliński, 2000] J. Zieliński. Inteligentne systemy w zarządzaniu – teoria i praktyka, PWN, Warszawa, 2000.
[Żurada, 1996] J.Żurada, M.Barski, W.Jędruch: Sztuczne sieci neuronowe, Warszawa: PWN, 1996, 375 pp.

Authors' Information

***Łukasz Paśko**, **M.Sc., Eng** – Rzeszow University of Technology, Department of Computer Science, The Faculty of Mechanical Engineering and Aeronautics, Powstancow Warszawy ave. 8, 35-959 Rzeszow, Poland;*
e-mail: lukasz.pasko48@gmail.com

Major Fields of Scientific Research: artificial intelligence, decision support systems, data mining.

Galina Setlak- **D.Sc, Ph.D., Associate Professor**, *Rzeszow University of Technology,* Department of Computer Science, *Powstancow Warszawy ave. 8, 35-959 Rzeszow, Poland,*
e-mail: gsetlak@prz.edu.pl

Major Fields of Scientific Research: decision-making in intelligent manufacturing systems, knowledge and process modeling, artificial Intelligence, neural networks, fuzzy logic, evolutionary computing, soft computing.

NEURAL NETWORKS, MACHINE LEARNING

THE EFFECT OF INTRODUCTION OF THE NON-LINEAR CALIBRATION FUNCTION AT THE INPUT OF THE NEURAL NETWORK

Piotr Romanowski

Abstract: *The paper presents the experiment on the time series whose elements are month values of BIS effective exchange rate of USD from January 1994 till March 2010. A tendency of BIS (Bank of International Settlements) effective exchange rate to increase or decrease is an expected value.*

First, a process of building of the neural network for events forecasting is presented, that is the selection of networks' architecture and parameters. Next, the effect of adding data calibrated by nonlinear input function to input data calibrated linearly is described. The nonlinear input function - hyperbolic tangent was accepted. Hyperbolic tangent sigmoid transfer function and log sigmoid transfer function are commonly used as transfer functions in neural networks.

Keywords: *neural network, time series.*

ACM Classification Keywords: *I.2.8 Data calibration.*

Introduction

Forecasting is one of the tools helping in taking decisions. The accuracy of forecasting, that is the agreement of the forecasted value with the real future actual value, is the measure of the correctness of the forecasting. New forecasting methods [Armstrong, 1992], [Kimberly, 2010], that will enable the increase of accuracy are the subject of many works. Artificial neural networks are commonly used tools for forecasting. Typical examples of prediction by the use of neural networks are areas of finance market, meteorology, medicine and many others. Many factors, such as the frequency with which data should be sampled, the number of data points

which should be used in the input representation, the time window size etc. influence the accuracy of prognosis. Moreover, the architecture and parameters of neural network are additional factors when neural networks are used [Gill,1978], [Graves, 2009], [Williams, 1989]. Figure 1 [Smith, 2003] presents typical network architecture.

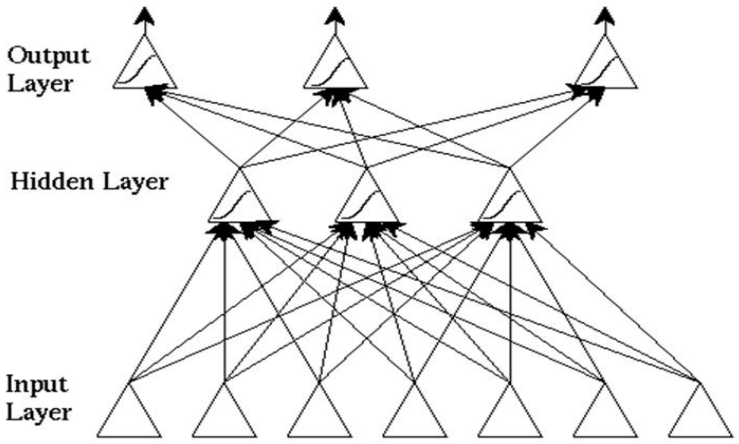


Fig. 1. Typical network architecture.

The process of neural network teaching is aimed at error minimization that is the minimization of differences between output signals and values of the training sample. During this process the optimal set of weights of individual neurons is searched. It is analogous to the process of minimizing of functional in a numerical solution of set of differential equations. The selection of architecture and parameters of the neural network and data preparation are similar to functional minimization, but more complicated.

Time series prediction

Time series are vectors, x(t), t=0,1,….n, where t represents consecutive time [Chen, 2002], [Oppenheim, 1999]. Work of neural networks above time series can be described as finding a function f: $R^N \rightarrow R$ such as to obtain an estimate of a x time t+d, from the N time steps back from time t, so that:

$$x(t+d)= f (x(t), x(t -1), ... , x(t - N +1)) \quad (1)$$

Experiment

Data

A subject of experiments was a set of average BIS effective exchange rate of USD from January 1994 till March 2010. It is worth mentioning that exchange rates are regarded as very difficult to predict [Stein, 1994]. In the paper, the whole set was named USD rate, its elements USD rate[i], i=1,…192 (the length of USD rate =192).

Usually, data calibration to an interval from 0 to 1 is the first step in data preparation. The following equations are used:

USDcalibrated = (USD rate –min(USDrate))/range;

Where range =max(USDrate)-min(USDrate)

Or for value from -1 to 1,

USDcalibrated=2*(USDrate-min(USDrate-range/2)/range (used in this work)

Figure 2. presents USD rate after calibration to the interval [-1; +1].

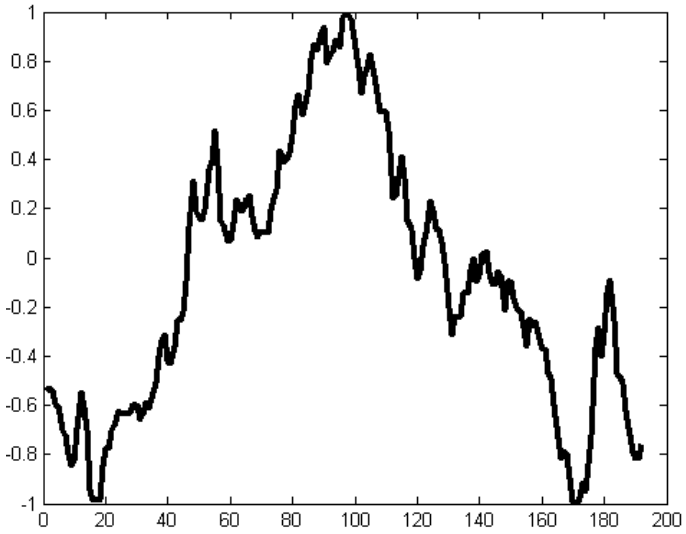


Fig. 2. USD rate after calibration.

The set of input elements was created according to equation (1). Accepting time window length N=15, 177 input vectors of N elements were obtained. A one layer input network (neuron number =N) and a one neuron output layer were accepted. Input layer neurons emit the output signal modified by a Log-sigmoid transfer function

logsig(n) = 1/(1+exp(-n)) [Matlab 7.1]. Output layer neurons emit a linear purelin(n) = n signal. Output functions are presented in Figure 3.

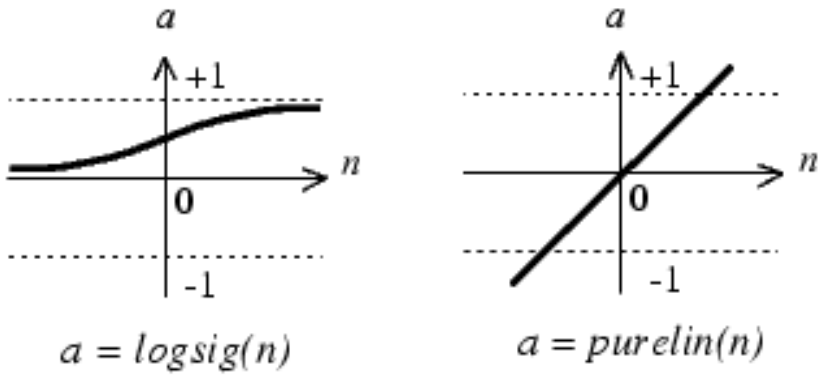


Fig. 3. Output functions.

A trainlm training function that updates weight and bias values according to Levenberg – Matquard optimization was used for network training [Lampton, 1997], [Marquardt, 1963].

Trainlim is often the fastest backpropagation algorithm in the MATLAB toolbox, and is highly recommended as a first choice supervised algorithm, although it does require more memory than other algorithms [Williams, 1989]. The network was trained and tested at the same set of vectors.

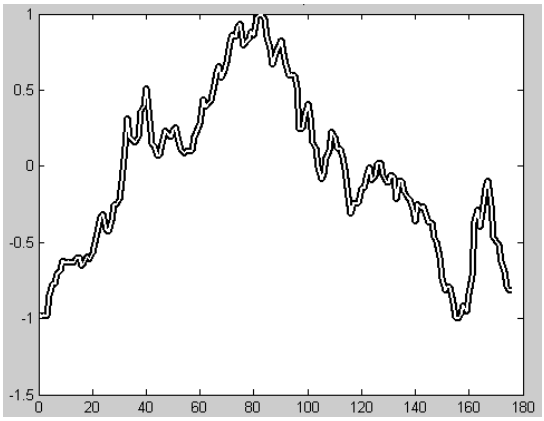


Fig.4. Output of the network can be seen as a thin, white line on the bold, black line which represents the measured data.

The following parameters were accepted:

The number of epochs = 100

Train Parameter goal = 1e-12 (error coefficient)

Size of time window = 15.

The network was trained well, values forecasted by the network agree with measured data (see Figure 4).

Mean absolute error of approximation was 4.56608e-005

Standard deviation of the approximation error was 0.000172702

Than, the set of input vectors was divided into a set of 160 training vectors and a set of 17 test vectors. After training, the results of prediction on the test set are not satisfactory (see Figure 5).

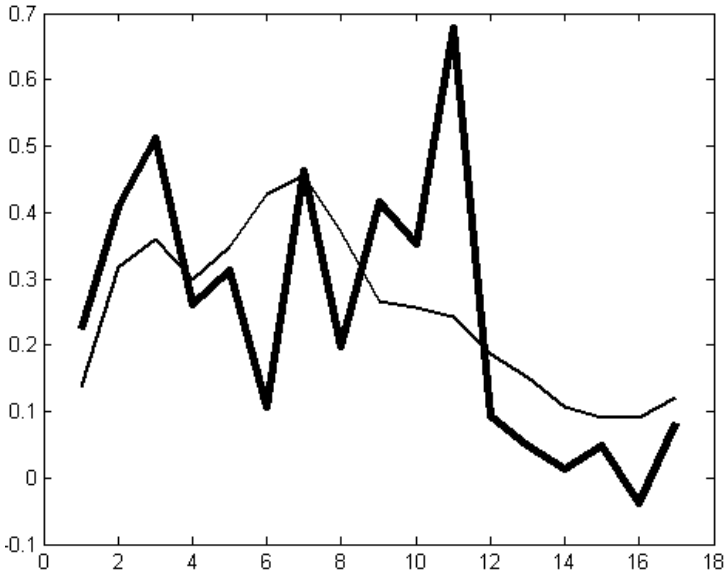


Fig. 5. The comparison of the real (thin line) and predicted (bold line) values of BIS effective exchange rate of USD.

Calculations were carried on for 100 epochs. Mean absolute approximation error was 52.8957 and standard deviation of the approximation error was 81.9857.

The proportion of the number of correct answers of the network to the number of test vectors was accepted as an efficiency of the network. The correct answer is the answer that has the same tendency as the test data value of USD exchange rate (1 - increase, 0 - else).

Next the set of 160 training vectors was divided into a set of 143 training vectors and 17 test vectors. It was assumed, that these 17 vectors are unknown.

Below, a short description of data preparation and selection of architecture and network parameters are presented.

Data preparation

The aim of the data preparation was establishing the optimal size of the time window and the length of the training vector establishing. 10 tests for each size of the time window from 1 to 20 were carried out, the obtained mean values are presented in Figure 6. Than a size of the time window equal 4 was accepted.

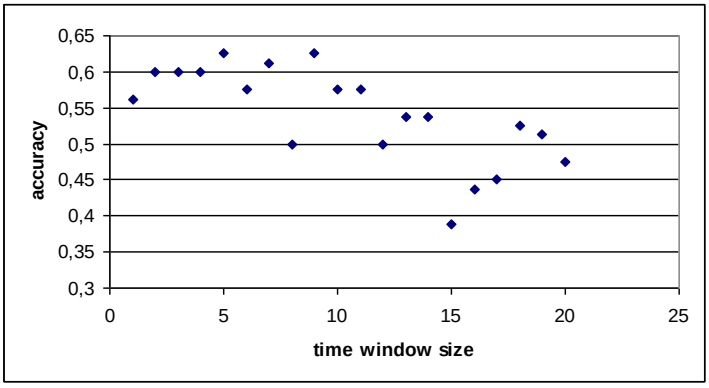


Fig.6. The accuracy of the network as a function of the time window size.

Next, the influence of the length of the training vector on the efficiency of the network was tested. Results are presented in Figure 7. For the further calculations the maximal length of the training vector equal 154 was accepted.

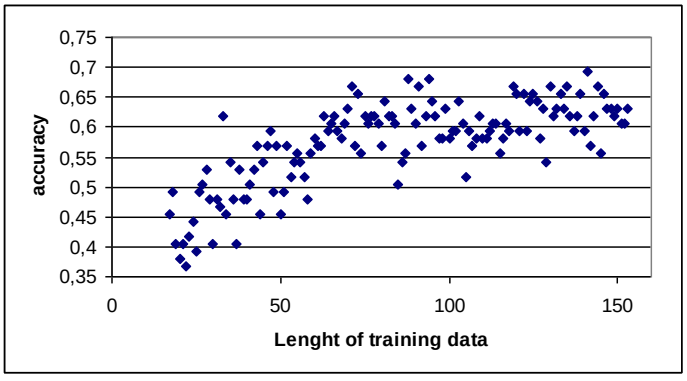


Fig.7. The accuracy of the network as a function of the length of the training vector.

In the paper the linear data calibration was accepted for choosing network's parameters and architecture. Data were projected into the [-1; +1] range, that means :

DATA calibrated = 2*(DATA-min(DATA)-range/2)/range.

Selection of architecture and network parameters.

Determination of the optimal number of training epochs is presented in Figure 8. Finally 50 training epochs were accepted.

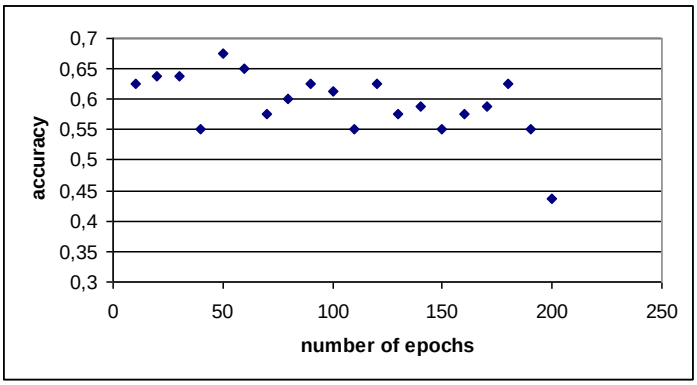


Fig.8. The influence of the number of the training epochs on the accuracy. Each point represents the mean value of 10 calculations

Consecutively, the results of the network extension of one hidden layer were investigated. Several kinds of layers with different number of neurons and different output functions were investigated, but as a mean accuracy always decreased, the idea of network extension was rejected. Figure 9 presents the result one of some calculations. Each point represents the mean value of 10 calculations.

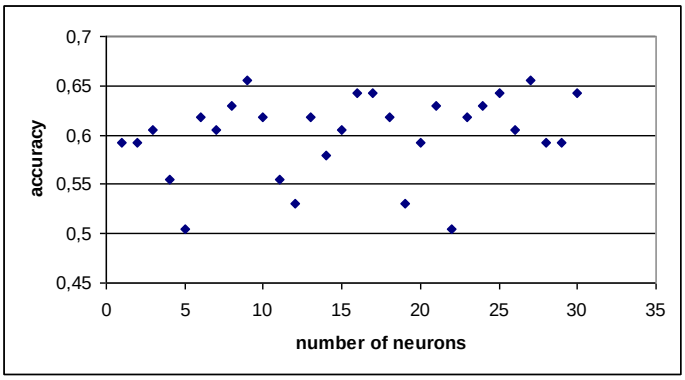


Fig. 9. The effect of introduction of the hidden layer with the changeable number of neurons.

Finally the log-sigmoid function was accepted for input layer and purelin for the output.

Neural network

Finally, the network with time size window = 4 (the number of neurons equals the length of the input vectors), with log-sigmoid transfer function and output layer with one neuron that emits the linear signal was accepted. The number of the training epochs equal 50 was accepted.

100 calculations for the accepted network were performed. The mean efficiency of the network was 0.6220. Figure 10. presents one of the plots that compare the network answers with the real values of the similar accuracy.

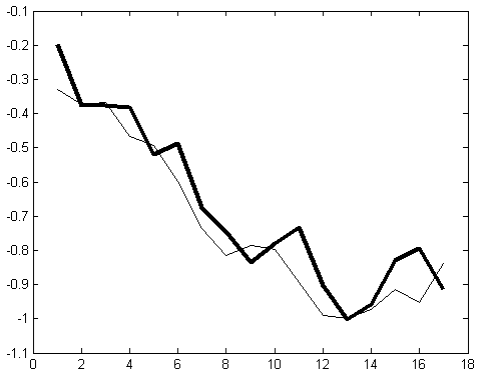


Fig. 10. The comparison of the real values of the BIS effective exchange rate of USD (the thin line) with the values calculated by the network (bold line)

Introduction of a non-linear calibration function

The input vectors included 4 consecutive mean month values of x=BIS effective exchange rate of USD calibrated to the interval [-1,+1]. The values calibrated by the non-linear function x' = tanh(beta*x) were added to them (see Figure 11).

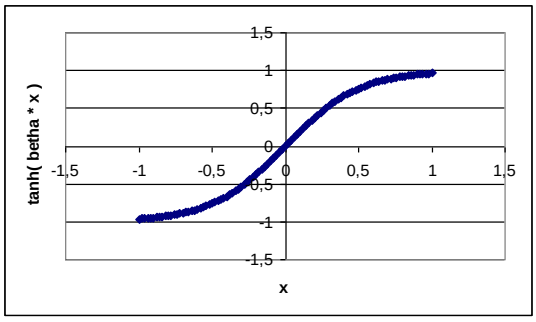


*Fig. 11. Non-linear calibration function x' = tanh(beta*x), beta = 2.0.*

The input vector includes now eight elements, instead of four. The input layer consists of eight neurons. Figure 12. presents results of such extension. Ten runs were performed for each value of beta coefficient from 0.5 to 3.5 (step 0.1). For each beta value of the accuracy increased.

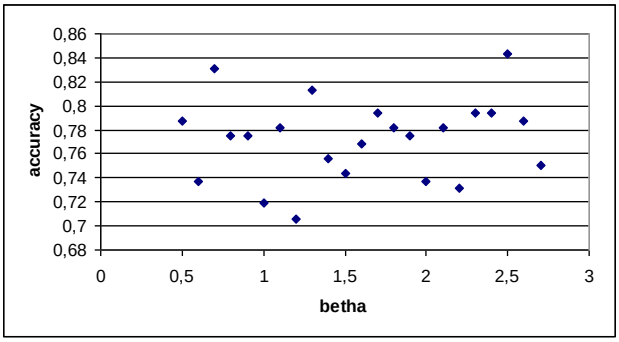


Fig. 12. The network accuracy after introduction of non-linear calibrated data.

Beta coefficient equal 2.0 was accepted.

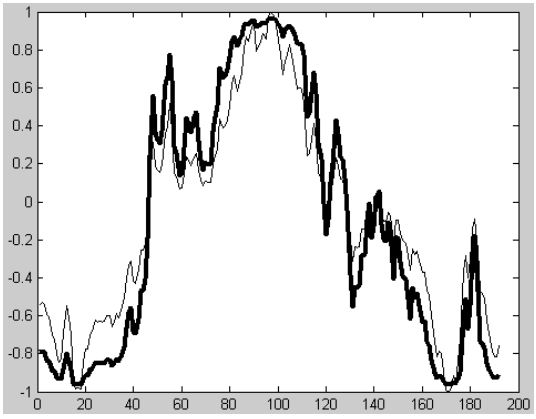


*Fig. 13. Data calibrated lineary - thin line and data calibrated by the use of non-linear function x' = tanh(beta*x) function - bold line.*

After 100 runs, the mean accuracy was 0.6740. This result is 5 per cent better that the accuracy for linear calibration (0.6220).

Two methods of calibration (linear and non- linear) were tested on 17 test vectors. After 100 runs, the mean accuracy was 0.8247 for linear calibration and 0.8667 for adding non-linear calibration. So, adding non-linear calibration increased the accuracy of 4.2 per cent.

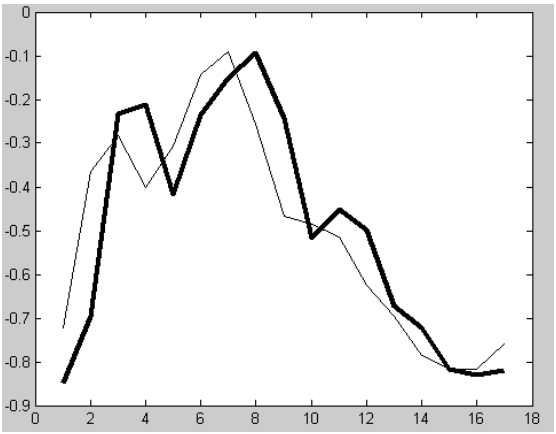


Fig. 14. Comparison of the network answer (bold line) with the real values (thin line) after one of 100 runs when the non – linear input function was used.

The test on the new set of input vectors gave higher accuracy. Probably, it is not due to the increase of the length of the training vector (see Figure 7.) but to the appearance of better recognized data.

Conclusions

The analyzed data are regarded as difficult to predict. On the other hand, there is a strong need of creating methods for forecasting the currency market. The significant difference of accuracy for different test vectors (65% and 85%) justify the statement, that one must be very careful using neural networks at the currency market.

The increase of the input vectors calibrated linearly with the values calibrated non – linearly caused the increase of accuracy by 4.6 per cent.

It seems, that results presented in this paper encourage further experiments and calculations.

Bibliography

[Armstrong , 1992] J.S.Armstrong, F.Collopy. Error Measures For Generalizing About Forecasting Methods: Empirical Comparisons. In: International Journal of Forecasting 8: 69–80, 1992.

[Chen, 2002] Y.Chen, G.Dong, J.Han, B.W.Wah, J.Wang. Multidimensional regression analysis of time-series data streams. In: Proc. 2002 Int. Conf. Very Large Data Bases (VLDB'02), 323-334, 2002.

[Gill,1978] P.E.Gill, W.Murray. Algorithms for the solution of the nonlinear least-squares problem. In: SIAM Journal on Numerical Analysis 15 (5): 977–992, 1978.

[Graves, 2009] A.Graves, J.Schmidhuber. Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks. Advances in Neural Information Processing Systems 22 In: NIPS'22, p 545-552, Vancouver, MIT Press, 2009.

[Kimberly, 2010] E.Kimberly. Production Planning and Inventory Control. In: McGraw-Hill. ISBN 0-412-03471-9, 2010.

[Lampton, 1997] M.Lampton. Damping-Undamping Strategies for the Levenberg-Marquardt Nonlinear Least-Squares Method. In: Computers in Physics Journal, 11(1):110–115, Jan./Feb, 1997.

[Marquardt, 1963] D.W.Marquardt. An algorithm for least-squares estimation of nonlinear parameters. In: Journal of the Society for Industrial and Applied Mathematics, 11:431–441, 1963.

[Oppenheim, 1999] A.V.Oppenheim, R.W.Schafer, J.A.Buck. Discrete-time signal processing. In: Upper Saddle River, N.J. Prentice Hall. pp. 468–471, 1999.

[Smith, 2003] L.Smith. An Introduction to Neural Networks, http://www.cs.stir.ac.uk/~lss/NNIntro/InvSlides.html. Centre for Cognitive and Computational Neuroscience, Department of Computing and Mathematics, University of Stirling. 2003.

[Stein, 1994] J.L.Stein. The Evolution of the Real Value of the U.S. Dollar Relative to the G7 Currencies. In: Ronald MacDonald. 1994.

[Williams, 1989] R.J.Williams. Complexity of exact gradient computation algorithms for recurrent neural networks. In: Technical Report Technical Report NU-CCS-89-27, Boston: Northeastern University, College of Computer Science, 1989.

[MATLAB 7.1] MATLAB Version 7.1 (R14SP3).

Authors' Information

Piotr Romanowski - Rzeszow University, Faculty of Mathematics and Natural Sciences / The Institute of Computer Science*. Al. Rejtana 16A, 35-310 Rzeszów, Poland; e-mail:* proman@univ.rzeszow.pl

Major Fields of Scientific Research: numerical methods, time series, artificial Intelligence

ADAPTIVE CLUSTERING OF INCOMPLETE DATA USING NEURO-FUZZY KOHONEN NETWORK

Yevgeniy Bodyanskiy, Alina Shafronenko, Valentyna Volkova

Abstract*: The clustering problem for multivariate observations often encountered in many applications connected with Data Mining and Exploratory Data Analysis. Conventional approach to solving these problems requires that each observation may belong to only one cluster, although a more natural situation is when the vector of features with different levels of probabilities or possibilities can belong to several classes. This situation is subject of consideration of fuzzy cluster analysis, intensively developing today.*

In many practical tasks of Data Mining, including clustering, data sets may contain gaps, information in which, for whatever reasons, is missing. More effective in this situation are approaches based on the mathematical apparatus of Computational Intelligence and first of all artificial neural networks and different modifications of classical fuzzy c-means (FCM) method.

But these methods are effective only in cases when the original data set is given beforehand and does not change during data processing. At the same time there is a wide class of problems when the data are fed to processing sequentially in on-line mode as it occurs in self-organizing Kohonen networks training. At the same time apriori it is not known which of the vectors-images contain gaps.

*In this paper the problem of probabilistic and possibilistic on-line clustering of data with gaps using Partial Distance Strategy is discussed and solved, self-organizing neuro-fuzzy Kohonen network and new self-learning algorithm that is the hybrid of "Winner-takes-more" rule and recurrent fuzzy clustering procedures are proposed and investigated***.***

Keywords*: Fuzzy clustering, Kohonen self-organizing network, learning rule, incomplete data with gaps.*

ACM Classification Keywords*: 1.2.6 [Artificial Intelligence]: Learning – Connectionism and neural nets; 1.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search – Control theory; 1.5.1 [Pattern Recognition]: Clustering – Algorithms.*

Data with gaps clustering on the basis of neuro-fuzzy Kohonen network

The clustering of multivariate observations problem often occurs in many applications associated with Data Mining. The traditional approach to solving these tasks requires that each observation may relate to only one cluster, although the situation more real is when the processed feature vector with different levels of probabilities or possibilities may belong more than one class. This situation is subject of fuzzy cluster analysis fast-growing at the present time [Bezdek, 1981; Hoeppner 1999; Xu, 2009].

However, in numerous problems of Data Mining, including, of course, clustering, the original data sets may contain gaps, information in which for some reasons, is missing. In this situation more effective are approaches based on the mathematical apparatus of computational intelligence, and especially, artificial neural networks [Marwala, 2009] and modifications of the classical c-means method (FCM) [Hathaway, 2001].

Notable approaches and solutions are efficient only in cases when the original array data set has batch form and does not change during the analysis. However there is enough wide class of problems when the data are fed to the processing sequentially in on-line mode as this occurs when training Kohonen self-organizing maps [Kohonen, 1995]. In this case, however, it is not known beforehand which of the processed vector-images contains gaps (missing values). This paper is devoted to solving the problem on-line clustering of data based on the Kohonen neural network, adapted for operation in presence of overlapping classes.

Adaptive algorithm for probabilistic fuzzy clustering

Baseline information for solving the tasks of clustering in a batch mode is the sample of observations, formed from N n-dimensional feature vectors $X = \{x_1, x_2, ..., x_N\} \subset R^n, x_k \in X, k = 1, 2, ..., N$. The result of clustering is the partition of original data set into m classes $(1 \le m \le N)$ with some level of membership $U_q(k)$ of k-th feature vector to the q-th cluster $(1 \le q \le m)$. Incoming data previously are centered and standardized by all features, so that all observations belong to the hypercube $[-1,1]^n$. Therefore, the data for clustering form array $\tilde{X} = \{\tilde{x}_1, ..., \tilde{x}_k, ..., \tilde{x}_N\} \subset R^n$, $\tilde{x}_k = (\tilde{x}_{k1}, ..., \tilde{x}_{ki}, ..., \tilde{x}_{kn})^T$, $-1 \le \tilde{x}_{ki} \le 1$, $1 < m < N$, $1 \le q \le m$, $1 \le i \le n$, $1 \le k \le N$.

Introducing the objective function of clustering [Bezdek, 1981]

$$E(U_q(k), w_q) = \sum_{k=1}^{N} \sum_{q=1}^{m} U_q^{\beta}(k) D^2(\tilde{x}_k, w_q)$$

with constraints $\sum_{q=1}^{m} U_q(k) = 1,\ 0 < \sum_{k=1}^{N} U_q(k) < N$ and solving the nonlinear programming problem, we get the probabilistic fuzzy clustering algorithm [Hoeppner, 1999; Xu, 2009]

$$\begin{cases} U_q^{(\tau+1)}(k) = \dfrac{(D^2(\tilde{x}_k, w_q^{(\tau)}))^{\frac{1}{1-\beta}}}{\sum_{l=1}^{m} (D^2(\tilde{x}_k, w_l^{(\tau)}))^{\frac{1}{1-\beta}}}, \\ w_q^{(\tau+1)} = \dfrac{\sum_{k=1}^{N} (U_q^{(\tau+1)})^{\beta} \tilde{x}_k}{\sum_{k=1}^{N} (U_q^{(\tau+1)}(k))^{\beta}}, \end{cases} \tag{1}$$

where w_q - prototype (centroid) of q-th cluster, $\beta > 1$ - parameter that is called fuzzyfier and defines "vagueness" the boundaries between classes, $D^2(\tilde{x}_k, w_q)$ - the distance between $\tilde{x}_k$ and w_q in adopted metric, $\tau = 0,1,2,\ldots$ - index of epoch information processing which is organized as a sequence of $w_q^{(0)} \to U_q^{(1)} \to w_q^{(1)} \to U_q^{(2)} \to \ldots$. The calculation process continues until satisfy the condition

$$\left\| w_q^{(\tau+1)} - w_q^{(\tau)} \right\| \le \varepsilon \ \forall\ 1 \le q \le m,$$

where ε - defines threshold of accuracy. Choosing $\beta = 2$ and taking the Euclidean distance, we get a popular algorithm of Bezdek's fuzzy c-means (FCM)

$$\begin{cases} U_q^{(\tau+1)}(k) = \dfrac{\left\| \tilde{x}_k - w_q^{(\tau)} \right\|^{-2}}{\sum_{l=1}^{m} \left\| \tilde{x}_k - w_l^{(\tau)} \right\|^{-2}}, \\ w_q^{(\tau+1)} = \dfrac{\sum_{k=1}^{N} (U_q^{(\tau+1)}(k))^2 \tilde{x}_k}{\sum_{k=1}^{N} (U_q^{(\tau+1)}(k))^2}. \end{cases}$$

The process of fuzzy clustering can be organized in on-line mode as sequentially processing. At this situation batch algorithm (1) can be rewritten in recurrent form [Bodyanskiy, 2005]

$$\begin{cases} U_q(k+1) = \dfrac{(D^2(\tilde{x}_{k+1}, w_q^{(k)}))^{\frac{1}{1-\beta}}}{\sum\limits_{l=1}^{m}(D^2(\tilde{x}_{k+1}, w_l(k)))^{\frac{1}{1-\beta}}}, \\ w_q(k+1) = w_q(k) + \eta(k+1)U_q^{\beta}(k+1)(\tilde{x}_{k+1} - w_q(k)), \end{cases} \tag{2}$$

(here $\eta(k+1)$ - learning rate parameter), which is a generalization of the clustering gradient procedure of Park-Dagher [Park, 1984] and the learning algorithm of Chung-Lee [Chung, 1994]. If the data are fed to the processing with high-frequency, recalculation of epochs is not made, if this frequency is low, between the instants k and $k+1$ it is possible to organize several epochs in an accelerated time.

It should be noted that the first expression in (2) can be rewritten in the form

$$U_q(k+1) = \frac{(D^2(\tilde{x}_k, w_q(k)))^{\frac{1}{1-\beta}}}{\sum\limits_{l=1}^{m}(D^2(\tilde{x}_k, w_l(k)))^{\frac{1}{1-\beta}}} =$$

$$= \frac{(D^2(\tilde{x}_k, w_q(k)))^{\frac{1}{1-\beta}}}{(D^2(\tilde{x}_k, w_q(k)))^{\frac{1}{1-\beta}} + \sum\limits_{\substack{l=1 \\ l\neq q}}^{m}(D^2(\tilde{x}_k, w_l(k)))^{\frac{1}{1-\beta}}} =$$

$$= \frac{1}{1 + (D^2(\tilde{x}_k, w_q(k)))^{\frac{1}{\beta-1}} \sum\limits_{\substack{l=1 \\ l\neq q}}^{m}(D^2(\tilde{x}_k, w_l(k)))^{\frac{1}{1-\beta}}},$$

for the Euclidean metric and $\beta = 2$ taking the form of the Cauchy function with a parameter of width σ^2:

$$U_q(k+1) = \frac{1}{1 + \dfrac{\left\|\tilde{x}_k - w_q(k)\right\|^2}{\sigma^2}},$$

$$\sigma^2 = (\sum_{\substack{l=1 \\ l\neq q}}^{m}\left\|\tilde{x}_k - w_l(k)\right\|^{-2})^{-1}.$$

This fact allows us to rewrite the second expression in (2) with $\beta = 2$ in the form

$$w_q(k+1) = w_q(k) + \eta(k+1)U_q^2(k+1)(\tilde{x}_{k+1} - w_q(k)) =$$
$$= w_q(k) + \eta(k+1)\varphi_q(k+1)(\tilde{x}_{k+1} - w_q(k))$$

where $U_q^2(k+1) = \varphi_q(k+1)$ - the bell-shaped neighborhood function of neuro-fuzzy Kohonen network [Gorshkov, 2009] designed for solving the fuzzy clustering task [Shafronenko, 2011] using the principle "winner-takes-more» (WTM).

Adaptive probabilistic fuzzy clustering algorithm for data with gaps

In the situation if the data in the array $\tilde{X}$ contain gaps, the approach discussed above should be modified accordingly. For example, in [Hathaway, 2001] it was proposed the modification of the FCM-procedure based on partial distance strategy (PDS FCM). Thus introducing, additional arrays:

$X_F = \{\tilde{x}_k \in \tilde{X} \mid \tilde{x}_k - vector\ containing\ all\ components\}$;

$X_P = \{\tilde{x}_{ki}, 1 \le i \le n, 1 \le k \le N \mid values\ \tilde{x}_k,\ available\ in\ \tilde{X}\}$;

$X_G = \{\tilde{x}_{ki} = ?, 1 \le i \le n, 1 \le k \le N \mid values\ \tilde{x}_k, absent\ in\ \tilde{X}\}$

and taking instead of the traditional Euclidean metric partial distance (PD):

$$D_P^2(\tilde{x}_k, w_q) = \frac{n}{\delta_{k\Sigma}} \sum_{i=1}^{n} (\tilde{x}_{ki} - w_{qi})^2 \delta_{ki},$$

the objective function of clustering

$$E(U_q(k), w_q) = \sum_{k=1}^{N} \sum_{q=1}^{m} U_q^{\beta}(k) \frac{n}{\delta_{k\Sigma}} \sum_{i=1}^{n} (\tilde{x}_{ki} - w_{qi})^2 \delta_{ki}$$

(here $\delta_{ki} = \begin{cases} 0 \mid \tilde{x}_{ki} \in X_G, \\ 1 \mid \tilde{x}_{ki} \in X_F, \end{cases}$

$\delta_{k\Sigma} = \sum_{i=1}^{n} \delta_{ki}$)

and solving nonlinear programming problem, we obtain the algorithm

$$\begin{cases} U_q^{(\tau+1)} = \dfrac{(D_P^2(\tilde{x}_k, w_q^{(\tau)}))^{\frac{1}{1-\beta}}}{\sum\limits_{l=1}^{m}(D_P^2(\tilde{x}_k, w_q^{(\tau)}))^{\frac{1}{1-\beta}}}, \\ w_{qi}^{(\tau+1)} = \dfrac{\sum\limits_{k=1}^{N}(U_q^{(\tau+1)}(k))^{\beta}\delta_{ki}\tilde{x}_{ki}}{\sum\limits_{k=1}^{N}(U_q^{(\tau+1)}(k))^{\beta}\delta_{ki}} \end{cases} \quad (4)$$

which is a generalization of the standard FCM-procedure (1).

Algorithm (4) can be rewritten in recurrent form

$$\begin{cases} U_q(k+1) = \dfrac{(D_P^2(\tilde{x}_{k+1}, w_q(k)))^{\frac{1}{1-\beta}}}{\sum\limits_{l=1}^{m}(D_P^2(\tilde{x}_{k+1}, w_q(k)))^{\frac{1}{1-\beta}}}, \\ w_{qi}(k+1) = w_{qi}(k) + \eta(k+1)U_q^{\beta}(k+1)(\tilde{x}_{k+1,i} - w_{qi}(k))\delta_{ki}, \end{cases} \quad (5)$$

with the second relation (5) that can be represented as learning algorithm for neuro-fuzzy Kohonen network:

$$w_q(k+1) = w_q(k) + \eta(k+1)\phi_q(k+1)(\tilde{x}_{k+1} - w_q(k)) \otimes \delta_k, \quad (6)$$

where $\phi_q(k+1) = U_q^{\beta}(k+1)$ - bell-shaped neighborhood function, $\delta_k = (\delta_{k1}, \ldots, \delta_{kn})^T$, $\otimes$ -symbol of direct product.

Thus, using a standard Kohonen network architecture and algorithm of its tuning (6) in on-line mode it is possible to solve the problem of fuzzy clustering data with gaps.

Adaptive algorithm for possibilistic fuzzy clustering

The main disadvantage of probabilistic algorithms is connected with the constraints on membership levels which sum has to be equal unity. This reason has led to the creation of possibilistic fuzzy clustering algorithms [Krishnapuram, 1993].

In possibilistic clustering algorithms the objective function has the form

$$E(U_q(k), w_q, \mu_q) = \sum_{k=1}^{N}\sum_{q=1}^{m} U_q^{\beta}(k) D^2(\tilde{x}_k, w_q) + \sum_{q=1}^{m}\mu_q \sum_{k=1}^{N}(1 - U_q(k))^{\beta} \quad (7)$$

where the scalar parameter $\mu \geq 0$ determines the distance at which level of membership equals to 0.5, i.e. if $D^2(\tilde{x}_k, w_q) = \mu_q$, then $w_q(k) = 0.5$.

Minimizing (7) relatively $U_q(k)$, w_q and μ_q we get the solution

$$\begin{cases} U_q^{(\tau+1)}(k) = \dfrac{1}{1+\left(\dfrac{D^2(\tilde{x}_k, w_q^{(\tau)})}{\mu_q^{(\tau)}}\right)^{\frac{1}{\beta-1}}}, \\ w_q^{(\tau+1)} = \dfrac{\sum\limits_{k=1}^{N}(U_q^{(\tau+1)}(k))^{\beta}\,\tilde{x}_k}{\sum\limits_{k=1}^{N}(U_q^{(\tau+1)}(k))^{\beta}}, \\ \mu_q^{(\tau+1)} = \dfrac{\sum\limits_{k=1}^{N}(U_q^{(\tau+1)}(k))^{\beta} D^2(\tilde{x}_k, w_q^{(\tau+1)})}{\sum\limits_{k=1}^{N}(U_q^{(\tau+1)}(k))^{\beta}}, \end{cases} \tag{8}$$

which with $\beta = 2$ and Euclidean metric has the form

$$\begin{cases} U_q^{(\tau+1)}(k) = \dfrac{1}{1+\dfrac{\left\|\tilde{x}_k - w_q^{(\tau)}\right\|^2}{\mu_q^{(\tau)}}}, \\ w_q^{(\tau+1)} = \dfrac{\sum\limits_{k=1}^{N}(U_q^{(\tau)}(k))^2\,\tilde{x}_k}{\sum\limits_{k=1}^{N}(U_q^{(\tau)}(k))^2}, \\ \mu_q^{(\tau+1)} = \dfrac{\sum\limits_{k=1}^{N}(U_q^{(\tau)}(k))^2\left\|\tilde{x}_k - w_q^{(\tau+1)}\right\|^2}{\sum\limits_{k=1}^{N}(U_q^{(\tau)}(k))^2}. \end{cases} \tag{9}$$

Information processing in the on-line mode (8), (9) can be written as [Bodyanskiy, 2005; Gorshkov, 2009]

$$\begin{cases} U_q(k+1) = \dfrac{1}{1+\left(\dfrac{D^2(\tilde{x}_{k+1}, w_q(k))}{\mu_q(k)}\right)^{\frac{1}{\beta-1}}}, \\ w_q(k+1) = w_q(k) + \eta(k+1)U_q^{\beta}(k+1)(\tilde{x}_{k+1} - w_q(k)), \\ \mu_q(k+1) = \dfrac{\sum\limits_{p=1}^{k+1} U_q^{\beta}(p) D^2(\tilde{x}_p, w_q(k+1))}{\sum\limits_{p=1}^{k+1} U_q^{\beta}(p)} \end{cases} \quad (10)$$

and

$$\begin{cases} U_q(k+1) = \dfrac{1}{1+\dfrac{\left\|\tilde{x}_k - w_q(k)\right\|^2}{\mu_q(k)}}, \\ w_q(k+1) = w_q(k) + \eta(k+1)U_q^{2}(k+1)(\tilde{x}_{k+1} - w_q(k)), \\ \mu_q(k+1) = \dfrac{\sum\limits_{p=1}^{k+1} U_q^{2}(p) \left\|\tilde{x}_p - w_q(k+1)\right\|^2}{\sum\limits_{p=1}^{k} U_q^{2}(p)}. \end{cases} \quad (11)$$

It's easily to see that relations (10), (11) are the Kohonen's self-learning WTM-rule with Cauchy functions as a neighborhood ones.

Adaptive algorithm for possibilistic fuzzy clustering of data with gaps

Adopting instead of Euclidean metric partial distance (PD), we can write the objective function of the type (7) as

$$E(U_q(k), w_q, \mu_q) = \sum_{k=1}^{N}\sum_{q=1}^{m} U_q^{\beta}(k) \frac{n}{\delta_{k\Sigma}} \sum_{i=1}^{n} (\tilde{x}_{ki} - w_{qi})^2 \delta_{ki} + \sum_{q=1}^{m} \mu_q \sum_{k=1}^{N} (1 - U_q(k))^{\beta}$$

and then solving the equations system

$$\begin{cases} \dfrac{\partial E(U_q(k), w_q, \mu_q)}{\partial U_q(k)} = 0, \\ \dfrac{\partial E(U_q(k), w_q, \mu_q)}{\partial \mu_q} = 0, \\ \nabla_{w_q} E(U_q(k), w_q, \mu_q) = \vec{0}, \end{cases}$$

get the procedure of type (8), which can be rewritten in the recurrent form

$$\begin{cases} U_q(k) = \dfrac{1}{1 + (\dfrac{D_P^2(\tilde{x}_{k+1}, w_q(k))}{\mu_q(k)})^{\frac{1}{\beta-1}}}, \\ w_{qi}(k+1) = w_{qi}(k) + \eta(k+1) U_q^{\beta}(k+1)(\tilde{x}_{k+1,i} - w_{qi}(k))\delta_{ki}, \\ \mu_q(k+1) = \dfrac{\sum\limits_{p=1}^{k+1} U_q^{\beta}(p) D_P^2(\tilde{x}_p, w_q(k+1))}{\sum\limits_{p=1}^{k+1} U_q^{\beta}(p)}. \end{cases}$$

The second relation can be rewritten as

$$w_q(k+1) = w_q(k) + \eta(k+1) U_q^{\beta}(k+1)(\tilde{x}_{k+1} - w_q(k)) \square \ \delta_k$$

coinciding with the learning procedure (6).

Thus, the process of fuzzy possibilistic clustering data with gaps can also be realized by using neuro-fuzzy Kohonen network.

Conclusion

The problem of probabilistic and possibilistic on-line fuzzy clustering of data with gaps based on the strategy of partial distances is considered. It is shown that it can be solved on the basis of self-organizing neuro-fuzzy Kohonen network. Proposed learning algorithm is a hybrid of rule "winner-takes-more" and recurrent fuzzy clustering algorithms.

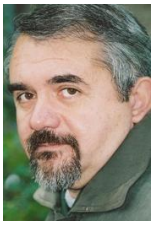
Bibliography

[Bezdek, 1981] J.C. Bezdek. Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York, 1981.

[Hoeppner, 1999] F Hoeppner, F. Klawonn, R. Kruse, T. Runker. Fuzzy Clustering Analysis: Methods for Classification, Data Analysis and Image Recognition. Chichester, John Wiley &Sons, 1999.

[Xu, 2009] R. Xu, D.C. Wunsch. Clustering. Hoboken, N.J. John Wiley & Sons, Inc., 2009.
[Marwala, 2009] T Marwala. Computational Intelligence for Missing Data Imputation, Estimation, and Management: Knowledge Optimization Techniques. Hershey-New York: Information Science Reference, 2009.
[Hathaway, 2001] R.J. Hathaway, J.C Bezdek. Fuzzy c-means clustering of incomplete data. IEEE Trans on Systems, Man, and Cybernetics, №5, 31, 2001, P. 735-744.
[Kohonen, 1995] T. Kohonen. Self-Organizing Maps. Berlin: Springer-Verlag, 1995.
[Bodyanskiy, 2005] Ye. Bodyanskiy. Computational intelligence techniques for data analysis. Lecture Notes in Informatics. Bonn: GI, 2005, V. P-72, P. 15-36.
[Park, 1984] D.C. Park, I. Dagher. Gradient based fuzzy c-means (GBFCM) algorithm. Proc. IEEE Int. Conf. on Neural Networks, 1984, P.1626-1631.
[Chung, 1994] F.L. Chung, T. Lee. Fuzzy competitive learning. Neural Networks, 1994, 7, №3, P.539-552.
[Gorshkov, 2009] Ye. Gorshkov, V. Kolodyazhniy, Ye. Bodyanskiy. New recursive learning algorithms for fuzzy Kohonen clustering network. Proc. 17th Int. Workshop on Nonlinear Dynamics of Electronc Systems. (Rapperswil, Switzerland, June 21-24, 2009) Rapperswil, Switzerland, 2009, P. 58-61.
[Shafronenko, 2011] A. Y. Shafronenko, V.V. Volkova, Ye. Bodyanskiy. Adaptive clustering data with gaps. Radioelectronics, informatics, control. – 2011. - №2. – P. 115-119 (in Russian)
[Krishnapuram, 1993] R. Krishnapuram, J.M. Keller. A possibilistic approach to clustering. Fuzzy Systems, 1993, 1, №2, P.98-110.

Authors' Information



Yevgeniy Bodyanskiy – Professor, Dr. – Ing. habil., Scientific Head of Control Systems Research Laboratory, Kharkiv National University of Radio Electronics, 14 Lenin Ave., Office 511, 61166 Kharkiv, Ukraine; e-mail: bodya@kture.kharkov.ua

Major Fields of Scientific Research: Artificial neural networks, Fuzzy systems, Hybrid systems of computational intelligence

Alina Shafronenko – intern-researcher of Control Systems Research Laboratory, Kharkiv National University of Radio Electronics, 14 Lenin Ave., Office 517, 61166 Kharkiv, Ukraine; e-mail: alinashafronenko@gmail.com

Major Fields of Scientific Research: neural networks, neural network processing of data with gaps, fuzzy clustering, clustering of data

Valentyna Volkova - Candidate of Technical Science (Ph.D.), Senior lecturer in Artificial Intelligence dept., Kharkiv National University of Radioelectronics Lenin Ave., 14, Kharkiv, 61166, Ukraine; e-mail: volkova@kture.kharkov.ua

Major Fields of Scientific Research: neural networks, fuzzy clustering, clustering of data

BIOINFORMATICS USING INTELLIGENT AND MACHINE LEARNING

A HYBRID INTELLIGENT CLASSIFIER FOR THE DIAGNOSIS OF PATHOLOGY ON THE VERTEBRAL COLUM

Essam Abdrabou

***Abstract**: The use of Machine Learning (ML) techniques is already widespread in Medicine Diagnosis. The use of these techniques helps increasing the efficiency of human diagnostic, which is significantly affected by the human conditions such as stress as well as the lack of experience. In this paper, integration between two ML techniques case-based reasoning (CBR) and artificial neural network (ANN) is used for the automation of the diagnosis of pathology on the vertebral column. CBR is used for indexing and retrieval. For adaptation, an untrained ANN is fed with the retrieved closest matches. Then the ANN is trained and queried with the new problem to give the adapted solution. Experiments are conducted on the vertebral column data set from University of California Irvine (UCI) machine learning repository. A comparison with several machine learning techniques used for classifying the same problem is performed. Results show that the hybridization between CBR and ANN helps in improving the classification.*

***Keywords**: Computer Aided Diagnosis System, Hybrid Intelligent Classifier, Vertebral Column, Case-Based Reasoning, Artificial Neural Network.*

***ACM Classification Keywords**: I.2.5 Expert system tools and techniques - Conference proceedings.*

Introduction

A hybrid intelligent system is one that combines at least two intelligent technologies. For example, combining a neural network with a fuzzy system results is a hybrid neuro-fuzzy system. Each component has its own strengths and weaknesses. Probabilistic reasoning is mainly concerned with uncertainty, fuzzy logic with imprecision, neural networks with

learning, and evolutionary computation with optimization. A good hybrid system brings the advantages of these technologies together [Negnevitsky, 2005].

Case-based Reasoning (CBR) is one of the fastest growing areas in the field of knowledge-based systems. It has been used to develop many systems applied in a variety of domains, including industry, design, law, medicine, and battle planning. CBR is based on psychological theories of human cognition [Watson, 1997]. It rests on the intuition that human expertise does not depend on rules or other formalized structures but on experiences. CBR claims to reduce the effort required for developing knowledge-based systems substantially compared with more traditional artificial intelligence approaches.

An artificial neural network (ANN) is a computational model that tries to simulate biological neural networks. Neural networks are used for performing classification and clustering tasks. A neural network consists of many simple interconnected processing units called neurons. The behavior of a neural network is determined by neuron interconnections and neuron parameters. Training data are used to train a neural network to perform its desired function. Various learning algorithms have been applied to train neural networks. Back propagation is the most well-known such algorithm [Haykin, 1999].

Hybridization between neural networks and CBR may develop extra advantages to both systems. On the one hand, neural networks provide efficiency, generalization and robustness that are important features in different domains. Meanwhile, classification and clustering functions are necessary in several CBR tasks. On the other hand, CBR offers openness and modularity to the integrated system by exploiting available cases [Prentzas & Hatzilygeroudis, 2009].

In this paper a combination of the CBR and ANN is done in order to develop an intelligent classifier for the pathology on the vertebral column. The vertebral column has several major functions. It encloses the spinal cord, that delicate bundle of nerve tissue which carries nerve impulses between the brain and the rest of the body. The vertebral column also provides structural support for the chest as well as the maintenance of the posture of the body and in movement. Injuries to the vertebral column are common cause severe pain in the injured area. In severe cases, the spinal cord may be affected as well [Seymour, 1998]. Thus, facilitating an auxiliary system to medical decision supporting is important.

The developed classifier is compiled using a domain-independent CBR shell designed and developed by the author which will be reference hereinafter as (eZ-CBR). This shell integrates the CBR and ANN in one application that facilitates the processing of different domain problems in few steps. Experiments are conducted using a dataset provided by University of California, Irvine ML repository [Frank & Asuncion, 2010].

This paper is organized in five sections. The first section is this introduction. The second section gives theoretical foundations about the vertebral column and related work and the combination between CBR and ANN. The third section discusses the eZ-CBR shell; it illustrates its architecture and some implementation issues. The fourth section presents the experimental work. It discusses different dataset attributes, the conducted experiments and compares the obtained results with previous obtained results from other intelligent diagnostic systems. Finally, the fifth section concludes the work.

Theoretical Foundations

Vertebral Column The spine (or backbone) is a column of 26 bones called vertebrae that extend in a line from the base of the skull to the pelvis. This is referred to as the "spinal column" or "vertebral column". The spinal column provides the main support for the upper body, allowing humans to stand upright or bend and twist, and it protects the spinal cord from injury [Seymour, 1998]. Figure 1 shows a front and side views of the vertebral column.

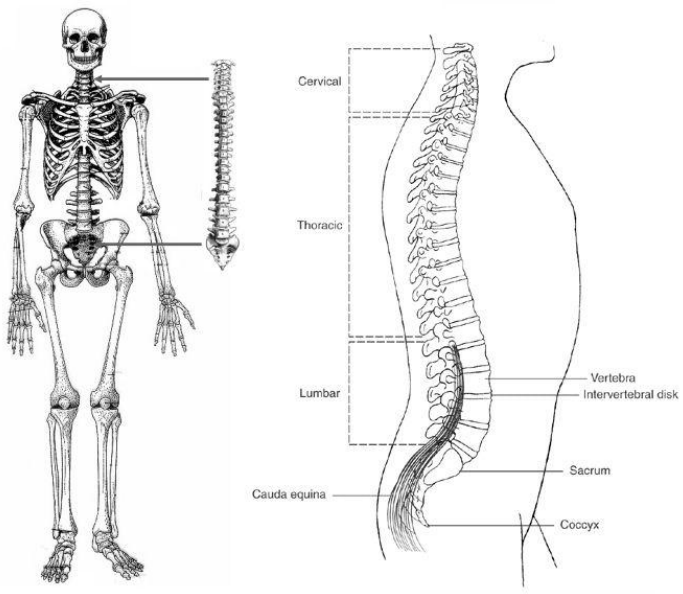


Fig. 1: The Vertebral Column

The lower part of the back holds most of the body's weight. Even a minor problem with the bones, muscles, ligaments, or tendons in this area can cause pain when a person stands, bends, or moves around. The disk which is a flat round plate like structure separates the bones that make up the spinal column. It is a fibrous structures filled with a pulpy,

gelatinous matter. It functions as shock absorbers for the spine. Disc-related injuries to the back can be associated with deformation of the discs, including bulging and rupturing of the discs. Less often, a problem with a disc can pinch or irritate a nerve from the spinal cord, causing pain that runs down the leg, below the knee.

The most common examples of pathologies of the vertebral column are disc hernia and spondylolisthesis that cause intense pain. A herniated disc may occur suddenly in an event such as a fall or an accident, or may occur gradually with repetitive straining of the spine. When a herniated disc occurs, the space for the nerves is further diminished, and irritation of the nerve results. Spondylolisthesis is a condition in which a break in both sides of the ring allows the body of the vertebra to slip forward. Spondylolisthesis results from repetitive extension of the back (bending backward). This causes weakness in the rings of the lumbar vertebrae, eventually leading to a break (fracture) in a ring [Lepori, 2011]. Figure 2 shows examples of pathologies of the vertebral column with focus on herniated disc and spondylolisthesis.

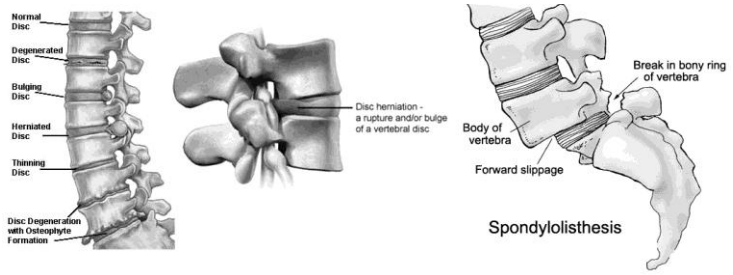


Fig. 2: Examples of Disk Problems

Related Work ML algorithms are already used in many medical diagnosis applications; however the use of them in diagnosis of Orthopedics is rare. This is due to the knowledge elicitation problem where there is a lack of clinical numeric values that describe the pathologies of the orthopedics. Some efforts have been explored in the following paragraphs which are based on the dataset provided by UCI [Frank & Asuncion, 2010] which is built by Dr. Henrique da Mota during a medical residence period in the Group of Applied Research in Orthopedics (GARO) of the Centre Médico-Chirurgical de Réadaptation des Massues, Lyon, France [Frank & Asuncion, 2010].

[Neto & Barreto, 2009] reported results from a performance comparison among some standalone ML algorithms Support Vector Machine (SVM), Multiple Layer Perceptron (MLP) and Generalized Regression Neural Network (GRNN) and their combinations in ensembles of classifiers. They used the same dataset to benchmark the performance. They evaluated the learning strategies in the classification modules according to their

ability in discriminating patients as belonging to one out of three categories: Normal, Disk Hernia and Spondylolisthesis.

[Mattos & Barreto, 2011] introduced two novel ensemble models built using Fuzzy Adaptive Resonance Theory (FA) and Self Organizing Map (SOM) Neural Networks as base classifier. They used the vertebral column dataset provided by UCI [Frank & Asuncion, 2010] to compare three proposed strategies that convert these two unsupervised learning to supervised learning to be applied for the vertebral column classification task. Choosing the appropriate parameters for training base classifiers, for each classification task, was tackled using a metaheuristic approach. The vertebral column dataset was one of ten datasets which were used as for comprehensive performance evaluation in order to compare the ART in Ensembles and Multiple SOM Classifiers in Ensembles variants built from standard supervised classifiers.

[Neto et al., 2011] incorporated the reject technique to the diagnosis of pathologies on the Vertebral Column. The reject option proposes a novel method to learn the reject region on complex data. They applied their technique on the same UCI dataset and compared it with several ML techniques.

Combination of CBR and ANN In case-based reasoning (CBR) systems expertise is embodied in a library of past cases, rather than being encoded in classical rules. Each case typically contains a description of the problem, plus a solution and/or the outcome. The knowledge and reasoning process used by an expert to solve the problem is not recorded, but is implicit in the solution [Aamodt & Plaza, 1994]. Whenever, a new input case has to be dealt with, the case-based system performs inference in four phases known as the CBR cycle: retrieve, reuse, revise and retain. Figure 3 summarizes the CBR cycle.

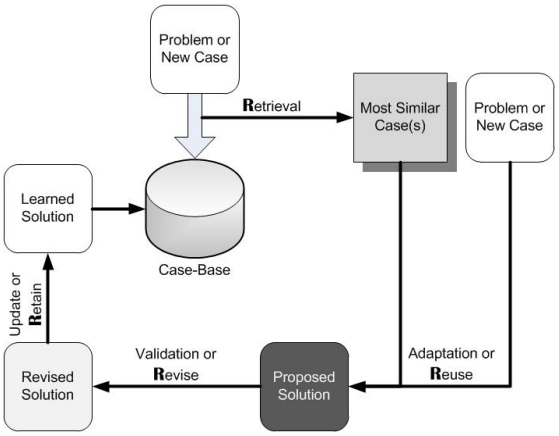


Fig. 3: The CBR Cycle

The retrieval phase retrieves from the case base the most relevant stored case or cases to the new case. The retrieval phase depends on indexing and similarity metrics. Indexing enables the efficient retrieval of relevant cases from the case base, thus limiting the search time. Similarity metrics assess the relevance of the retrieved cases to the new case. A simple approach to similarity assessment is the nearest neighbor matching [Kolodner, 1993]. Weights may be assigned to case features to denote feature importance in similarity assessment.

Adapting the most relevant retrieved case to meet the requirements of the new case is an important process due to the fact that retrieval involves partial matching. Adaptation focuses on differences between the most relevant case and the new case. Various adaptation methods have been developed such as substitution, transformation and derivational replay [Kolodner, 1993]. Adaptation can be a complex and time-consuming task usually requiring domain-dependent knowledge and sometimes user intervention [Kolodner, 1993]. There are some developed techniques have been developed to automatically acquire adaptation knowledge. Other techniques decrease the need for adaptation by retrieving cases that are easier to adapt. However, case adaptation is in many ways the Achilles' heel of CBR [Watson, 1997].

Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. It is composed of massively parallel computing systems consisting of an extremely large number of simple processors with many interconnections. ANN models attempt to use some organizational principles believed to be used in the human [Bishop, 1995]. The back-propagation algorithm has emerged as the workhorse for the design of a special class of layered feed-forward networks known as multilayer perceptrons (MLP). A multilayer perceptron has an input layer of source nodes and an output layer of neurons (i.e., computation nodes); these two layers connect the network to the outside world. In addition to these two layers, the multilayer perceptron usually has one or more layers of hidden neurons, which are so called because these neurons are not directly accessible. The hidden neurons extract important features contained in the input data [Bishop, 1995]. Neural networks also have some disadvantages such as the required training time may be extensive and convergence to an acceptable solution is not always assured, the initialization of weights may play an important role in the training process leading to different solutions, and the determination of neural network topology (such as finding the required number of hidden nodes) is done on a trial-and error approach [Prentzas & Hatzilygeroudis, 2009].

Neural networks are usually combined with CBR to perform tasks such as indexing, retrieval and adaptation. In this way, appealing characteristics of neural networks such as

parallelism, robustness, adaptability, generalization and ability to cope with incomplete input data are utilized.

eZ-CBR Shell

eZ-CBR shell is designed using object-oriented paradigm. So the entire shell is consisted of interacting objects which are grouped into three main parts the input part, the processing part which deals with the CBR process, and the output part. The input part deals with domain definition and loading different files required for building a CBR application. The CBR process part has all the necessary classes and functions required to complete the CBR process. The output part is responsible for writing the output.

Case Representation eZ-CBR shell applies object-oriented techniques for representing cases [Bergmann et al., 2005]. Such representations are particularly suitable for complex domains in which cases with different structures occur. The case in eZ-CBR shell is represented by a list of attributes. The list is dynamically allocated so the case can be represented with any number of attributes. The attribute type is lately bound to its actual type using polymorphism. The case itself can be one of the attributes in the attribute list. Figure 4 shows the case structure class diagram along with different classes that constitute the case class.

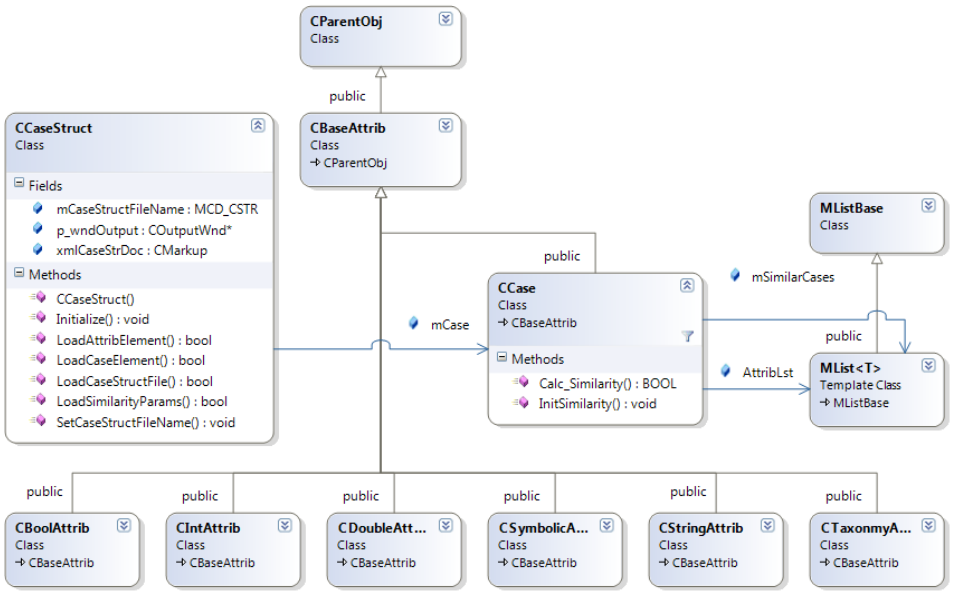


Fig. 4: The Case Structure Class Diagram in eZ-CBR Shell

Similarity is always used for describing something like "closely related". Similarity measures for such object-oriented representations are often defined by the general scheme: The goal is to determine the similarity between two objects, i.e., one object representing the case (or a part of it) and one object representing the query (or a part of it). This similarity is called object similarity (or global similarity). The object similarity is determined recursively in a bottom up fashion, i.e., for each simple attribute, a local similarity measure determines the similarity between the two attribute values, and for each relational slot an object similarity measure recursively compares the two related sub-objects. Then the similarity values from the local similarity measures and the object similarity measures, respectively, are aggregated (e.g., by a weighted sum) to the object similarity between the objects being compared [Bergmann & Stahl, 1998]. The similarity between a query, I and a case, J of a class C is defined as the sum of the similarities of its constituent features multiplied by their relevance weights as described in Equation 1.

$$Sim_C(I,J) = \sum_{i=0}^{n} w_i \times sim_i(I_i,J_i) \quad with \sum_{i=0}^{n} w_i = 1 \tag{1}$$

Where w_i is the feature relevance weight and sim_i is the local similarity measure (i.e. feature specific similarity measure).

Case Retrieval eZ-CBR shell uses similarity based retrieval with K-Nearest Neighbor algorithm. The following is the retrieval steps employed by the process.

- For a given a query case instance, calculate the distance between the query-instance and all the case-base bases using the similarity measures as described before;
- Sort the distance and determine nearest neighbors based on the minimum distance;
- Determine parameter K = number of nearest neighbors. The K value is configured by the user;
- Gather the category of the nearest neighbors;
- Use simple majority of the category of nearest neighbors as the prediction value of the query instance.

Adaptation eZ-CBR shell employs feed-forward back-propagation artificial neural network [Bishop, 95] to adapt the retrieved cases to the solution. The following is a list of steps during the adaptation process that are required to be conducted for each query case.

- For each query case, get the most similar cases from the retrieval process;

- Prepare the attributes of the most similar cases to feed them as input for an initialized feed-forward back-propagation neural network;
- Configure the neural network topology by determining the number of hidden layers and number of neurons in each layer;
- Train the network using back propagation;
- When converging is done, query the network with the query case;
- Collect the neural network output which is the adapted solution for the selected query;
- Repeat for other queries.

Configuration of the neural network is done through configuration parameters which define the number of hidden layers and number of neurons in each layer in addition to the training parameters such as learning rate, momentum, maximum number of epochs and the desired accuracy. The number of the hidden layers and the number of neurons in each layer are set to an arbitrary value at the beginning, then to find out how many hidden neurons are required trial and error approach can be employed.

The Vertebral Column Experiments

UCI Vertebral Column Data Set The data set has been organized in two different but related classification tasks. The first task is for classifying patients as belonging to one out of three categories: Normal, Disk Hernia or Spondylolisthesis. For the second task, the categories Disk Hernia and Spondylolisthesis were merged into a single category labeled as Abnormal. Table 1 shows the two tasks.

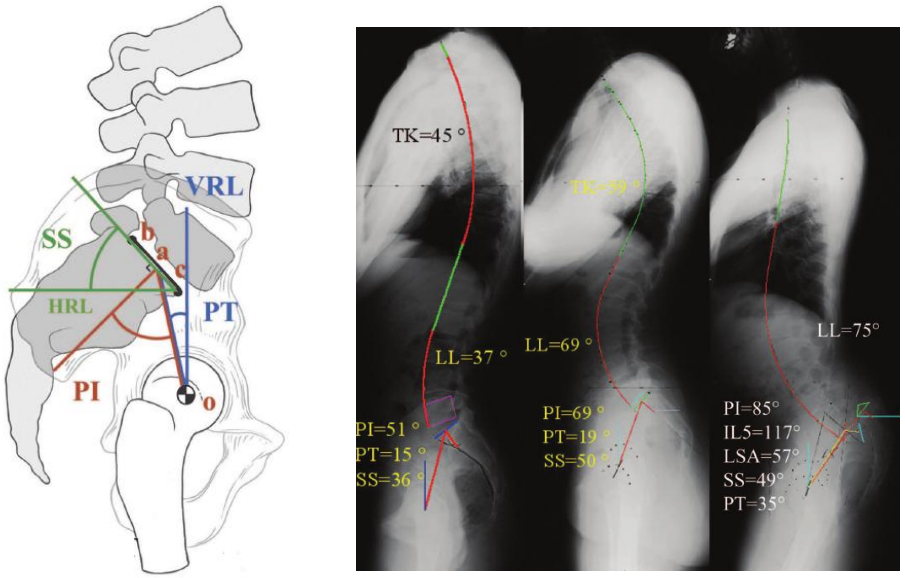
Table 1: UCI Vertebral Column Data Set Classification

Task	Task 1			Task 2	
Classification	Normal (NO)	Disk Hernia (DH)	Spondylolisthesis (SL)	Normal (NO)	Abnormal (AB)
No. of Patients	100	60	150	100	210

Each patient is represented in the data set by six biomechanical attributes derived from the shape and orientation of the pelvis and lumbar spine (in this order): pelvic incidence, pelvic tilt, lumbar lordosis angle, sacral slope, pelvic radius and grade of spondylolisthesis.

Figure 5A [Labelle et al., 2005] describes graphically some of the above attributes. Pelvic incidence (PI) is defined as an angle subtended by line (oa), which is drawn from

the center of the femoral head to the midpoint of the sacral endplate and a line perpendicular to the center of the sacral endplate. The sacral endplate is defined by the line segment (bc) constructed between the posterior superior corner of the sacrum and the anterior tip of the endplate at the sacral promontory. For the case when the femoral heads are not superimposed, the center of each femoral head is marked, and a connecting line segment will connect the centers of the femoral heads. The pelvic radius will be drawn from the center of this line to the center of the sacral endplate. Sacral slope (SS) is defined as the angle subtended by a horizontal reference line (HRL) and the sacral endplate line (bc). Pelvic tilt (PT) is defined as the angle subtended by a vertical reference line (VRL) originating from the center of the femoral head (o) and the pelvic radius (oa). It is positive when the hip axis lies in front of the middle of the sacral endplate [Labelle et al., 2005]. Lordosis angle is the bigger sagittal angle between the sacrum superior plate and the lumbar vertebra superior plate or thoracic limit. The grade of spondylolisthesis is the percentage level of slipping between the inferior plate of the fifth lumbar vertebra and the sacrum [Neto et al., 2011].



(A) The Pelvic Morphology Parameters

(B) Illustrative Cases of two Normal Adults and one subject with Spondyloptosis

Fig. 5 [Labelle et al., 2005]

The association between PI and spondylolisthesis has been reported in many cases [Labelle et al., 2005]. As shown in Figure 5B, the normal case on the left has a pelvic incidence angle within normal limits, while the normal case in the middle has a PI value

above normal. The case on the right has a developmental spondyloptosis. It can be clearly seen that the normal spine adjusts to pelvic morphology: the greater the PI, the greater will be SS and/or PT, and consequently, the greater will be the lumbar lordosis as the spine adjusts to maintain a stable posture [Labelle et al., 2005].

The Experiments Two major experiments have been conducted to automatically classify patients. Each major experiment represents a task. So, eZ-CBR has built one ternary classifier for task 1 and one binary classifier for task 2. In both experiments the data set has been randomly split into two parts one used for training and one for testing the results. Also, the classifier is tested when the entire data set has been used for training and the same number of queries has been tested. Each experiment has its own retrieval and adaptation parameters. Some of ANN learning parameters have been adjusted by trial and error, and they remain constant in all runs of both experiments. These parameters are the learning rate (0.9), the momentum (0.7), the maximum number of epochs (25,000) and the mean square error (0.0001). Table 2 and Table 3 show different parameters for each major experiment for task 1 and task 2 respectively.

Table 2: Task 1 Automatic Ternary Classification Experiment

Runs		**Run 1**	**Run 2**	**Run 3**	**Run 4**
Number of training cases		270	310	230	310
Number of Query cases		40	40	80	80
Training : Testing		13 : 87	N/A	25 : 75	N/A
Retrieval K Nearest Neighbor		40	40	30	40
ANN Adaptation	No. of Neurons in Input Layer	6	6	6	6
	No. of Neurons in Output Layer	3	3	3	3
	No. of Neurons in Hidden Layer(s)	6, 6	6, 6	6, 6	6, 6
Accuracy %		±85%	±90 %	±84%	±88%

In experiment 1, where the classifier seeks three outputs normal, disk herniated, or spondyloptosis the used neural network had two hidden layers each one had six neurons while the output layer had three neurons each one was responsible for one class. In run 1 and 3, the data set was randomly split into two parts the largest part was used for the training of the eZ-CBR while the smallest part was used in testing. The number of K closest matches was set to an arbitrary value also obtained by trial and error. The level

of accuracy gained in these runs was approximately 85%. In run 2 and run 4, the entire data set was used for training and the same testing part was used. In these runs, where exact match in the retrieval process is likely to happen, the obtained accuracy was approximately 89%.

Table 3: Task 2 Automatic Binary Classification Experiment

Runs		**Run 1**	**Run 2**	**Run 3**	**Run 4**
Number of training cases		270	310	230	310
Number of Query cases		40	40	80	80
Training : Testing		13 : 87	N/A	25 : 75	N/A
Retrieval K Nearest Neighbor		60	60	60	60
ANN Adaptation	No. of Neurons in Input Layer	6	6	6	6
	No. of Neurons in Output Layer	1	1	1	1
	No. of Neurons in Hidden Layer(s)	6	6	6	6
Accuracy %		±85%	±98%	±86%	±93%

In experiment 2, the used neural network only had one neuron in the output layer responsible for the binary classification. Only one hidden was used. The same splitting of the data set has been performed like experiment 1. In run 1 and 3, where a part of the data set was used in the training, the accuracy obtained was approximately 86% while in run 2 and 4 where the entire data set was used in the training; the obtained accuracy was approximately 95%.

Results Comparison [Neto & Barreto, 2009] reported results from a performance comparison among some standalone ML algorithms Support Vector Machine (SVM), Multiple Layer Perceptron (MLP) and Generalized Regression Neural Network (GRNN) the accuracy obtained was 82%, 83%, and 75% for each of the used algorithms respectively. After ensemble these classifiers they become C-SVM, C-MLP and C-GRNN, and reached 94%, 88%, and 81%. [Mattos & Barreto, 2011] tested the same data set on several developed ensemble classifiers built using built using Fuzzy Adaptive Resonance Theory (FA) and Self Organizing Map (SOM) Neural Networks as base classifier. Average accuracy obtained during their experiments was approximately 83%. [Neto et al., 2011] incorporated the reject technique to classifiers based on SVM with different kernels, and they could reach average approximate accuracy of 85%.

Excluding the high accuracy obtained from eZ-CBR during the experiments in which the entire data set was used of the training, eZ-CBR obtained average approximate accuracy is 85% which is almost the same accuracy obtained from other ML techniques.

Conclusion

In this paper a hybrid CBR and ANN classifier is developed for the classification of the pathology on vertebral column. The application is developed using eZ-CBR shell. eZ-CBR shell is a hybrid case-based reasoning and neural network tool that is developed by the author. The developed classifier is successful up to ±85% in classification of abnormal Pelvic Morphology patients. The obtained accuracy is almost the same accuracy obtained by other researchers who classified the same data set using other ML algorithms.

eZ-CBR shell shows a great potential in the hybridization between CBR and NN systems. CBR and NN are similar in that they perform the same kind of processing: given a problem, finding a solution with respect to the previous problems encountered. In the case of the CBR, this is done with a step-by-step symbolic method whereas in the case of the NNs, this is done with some numeric method. But, from an external point of view, the processes remain essentially the same. CBR and ANN are complementary on several points. On the kind of data they can handle, CBR deals easily with structured and complex symbolic data while ANN deal easily with numeric data. Therefore, a system able to deal with both kinds of representations would be suitable. On the way the problem space is represented, it is often difficult for a neural network to learn special cases, because of an over-generalization. On the opposite, a CBR system can easily deal with these special cases. Thus a combined system shows good generalization capabilities.

As for future research, an automated topology configurator needs to be added in the eZ-CBR shell in the adaptation part. Instead of adjusting ANN topology and learning parameters using trial and error technique, another ML may be incorporated to automatically optimize the ANN topology and learning parameters. Such ML algorithm may be an evolutionary algorithm that will be able to search for the optimum ANN topology without users' intervention.

Bibliography

[Aamodt & Plaza, 1994] A.Aamodt and E.Plaza. Case-Based Reasoning: Foundational Issues, Methodological Variation and System Approaches. AICOM, Vol. 7, No. 1, pp. 39-58, 1994.

[Bergmann et al., 2005] R.Bergmann, J.Kolodner and E.Plaza. Representation in case-based reasoning. Engineering, 00, 1-4.

[Bergmann & Stahl, 1998] R.Bergmann and A.Stahl. Similarity measures for object-oriented case representations. Advances in Case-Based Reasoning, No. 1488, Springer-Verlag London, UK, pp. 37-44, 1998.

[Bishop, 1995] C.M.Bishop. Neural Networks for Pattern Recognition. Oxford University Press, New York, 1995.

[Frank & Asuncion, 2010] A.Frank and A.Asuncion. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml Irvine, CA: University of California, School of Information and Computer Science, 2010.

[Haykin, 1999] S.Haykin, Neural Networks: A Comprehensive Foundation. 2nd Ed, Englewood Cliffs, NJ: Prentice-Hall., 1999.

[Kolodner, 1993] J.L.Kolodner. Case-Based Reasoning, California: Morgan Kaufmann Publishers.

[Labelle et al., 2005] H.Labelle, P.Roussouly, E.Berthonnaud, J.Dimnet and M.O'Brien. The Importance of Spino-Pelvic Balance in L5–S1 Developmental Spondylolisthesis: A Review of Pertinent Radiologic Measurements. SPINE Volume 30, Number 6S, pp S27–S34. Lippincott Williams & Wilkins, Inc, 2005.

[Lepori, 2011] L.R.Lepori. Diseases of the vertebral column Miniatlas. Letbar Asociados S.A, 2011.

[Mattos & Barreto, 2011] C.L.C.Mattos and G.A.Barreto. ARTIE and MUSCLE models: building ensemble classifiers from fuzzy ART and SOM networks. Neural Computing & Applications, pp. 1-13, October 2011.

[Negnevitsky, 2005] M.Negnevitsky. Artificial intelligence: a guide to intelligent systems. 2nd Ed. Addison-Wesley, pp. 259-299, 2005.

[Neto & Barreto, 2009] A.R.R.Neto and G.A.Barreto. On the application of ensembles of classifiers to the diagnosis of pathologies of the vertebral column: A comparative analysis. IEEE Transactions on Latin America 7(4), 487-496 (Aug 2009).

[Neto et al., 2011] A.R.R.Neto, R.Sousa, G.A.Barreto, and J.S.Cardoso. Diagnostic of pathology on the vertebral column with embedded reject option. In Proceedings of the 5th Iberian conference on Pattern recognition and image analysis (IbPRIA'11), Jordi Vitrià, João Miguel Sanches, and Mario Hernández (Eds.). Springer-Verlag, Berlin, Heidelberg, 588-595.

[Prentzas & Hatzilygeroudis, 2009] J.Prentzas and I.Hatzilygeroudis. Combinations of case-based reasoning with other intelligent methods. International Journal of Hybrid Intelligent Systems 6, 189–209, 2009.

[Seymour, 1998] S.Seymour. Bones: Our Skeletal System (Human Body). New York: Morrow (Harper-Collins), 1998.

[Watson, 1997] I.Watson. Applying Case-Based Reasoning: Techniques for Enterprise Systems. California: Morgan Kaufmann Publishers, 1997.

Authors' Information

Essam Abdrabou *– Adjunct Assistant Professor, Faculty of Computer Science, October University of Modern Sciences and Arts, 26 July Mehwar Road intersection with Wahat Road, 6 October City, Egypt; e-mail: essamamin@gmail.com*

Major Fields of Scientific Research: Artificial intelligence, Expert systems and Case-Based Reasoning especially in the design domain.

SUPPORT VECTOR MACHINES FOR CLASSIFICATION OF MALIGNANT AND BENIGN LESIONS

Anatoli Nachev, Mairead Hogan

Abstract: *This paper presents an exploratory study of the effectiveness of support vector machines used as a tool for computer-aided breast cancer diagnosis. We explore the discriminatory power of heterogeneous mammographic and sonographic descriptors in solving the classification task. Various feature selection techniques were tested to find a set of descriptors that outperforms those from similar studies. We also explored how choice of the SVM kernel function and model parameters affect its predictive abilities. The kernels explored were linear, radial basis function, polynomial, and sigmoid. The model performance was estimated by ROC analysis and metrics, such as true and false positive rates, maximum accuracy, area under the ROC curve, partial area under the ROC curve with sensitivity above 90%, and specificity at 98% sensitivity. Particular attention was paid to the latter two as lack of specificity causes unnecessary surgical biopsies. Experiments registered that an appropriate reduction of variables can greatly improve the predictive power of the model, as long as the choice of the kernel affects the model performance marginally. We also found that the SVM is superior to the common classification technique used in the field - MLP neural networks.*

Keywords: *data mining, support vector machines, heterogeneous data; breast cancer diagnosis, computer aided diagnosis.*

ACM Classification Keywords: I.5.2- Computing Methodologies - Pattern Recognition – Design Methodology - Classifier design and evaluation.

Introduction

A proper treatment of breast cancer disease requires timely, reliable, and accurate diagnosis, which allows radiologists and physicians to differentiate between benign and malignant lesions. Many computer-aided detection/diagnosis (CAD) tools currently support medical practices by capturing knowledge from previous cases and applying that knowledge to the new cases. CAD is a typical machine-learning problem, which has been dealt with by various data mining techniques and tools such as linear discriminant analysis (LDA), logistic regression analysis (LRA), multilayer perceptions (MLP), etc. [Chen et al., 2009].

Most of the current implementations tend to use only one information source, usually mammographic data in the form of data descriptors defined by the Breast Imaging Reporting and Data System (BI-RADS) lexicon, developed by the American College of Radiology (ACR) in order to standardize the mammographic language and interpretations, and to facilitate communication between clinicians [BI-RADS, 2003], [Kopans, 1992]. Jesneck et al. [2007] have used a novel combination of BI-RADS mammographic and sonographic descriptors and some proposed by Stavros et al. [1995] in order to build a predictive model based on MLP, which shows superior characteristics to those that use one data source. Our study takes that approach, but investigate another predictive technique - support vector machines (SVM). We also address the problem of high false positive rate of indication for biopsy (specificity rate), which causes unnecessary surgical biopsies, lowers the efficiency of the diagnosis, exposes patients to discomfort, and creates financial burden as procedures cost thousands of euros each [Lacey et al., 2002].

Further to the study of Jesneck et al. [2007] who used a set of fourteen descriptors to train and test a MLP neural network, we explore the discriminatory power of all descriptors in order to seek alternative sets that when applied to SVM can ensure even higher sensitivity and specificity.

The paper is organized as follows: Section 2 provides a brief overview of the support vector machines used as data mining tools; Section 3 introduces the dataset used in this study and discusses variable selection as part of the data preprocessing; Section 4 presents and discuses results obtained from experiments; and Section 5 gives the conclusions.

Support Vector Machines

Support vector machines are common machine learning techniques. They belong to the family of generalized linear models, which achieve a classification or regression decision based on the value of the linear combination of input features. Using historical data along with supervised learning algorithms, SVM generate mathematical functions to map input variables to desired outputs for classification or regression prediction problems.

SVM, originally introduced by Vapnik [1995], provide a new approach to the problem of pattern recognition with clear connections to the underlying statistical learning theory. They differ radically from comparable approaches such as neural networks because SVM training always finds a global minimum in contrast to the neural networks. SVM can be formalized as follows. Training data is a set of points of the form

$$D = \{(x_i, c_i) \mid x_i \in \Re^p,\ c_i \in \{-1, 1\}\}_{i=1}^{n}, \quad (1)$$

where the c_i is either 1 or -1, indicating the class to which the point x_i belongs. Each data point x_i is a p-dimensional real vector. During training a linear SVM constructs a p-1-dimensional hyperplane that separates the points into two classes (see Figure 1). Any hyperplane can be represented by: $w \cdot x - b = 0$ where w is a normal vector and $\cdot$ denotes dot product. Among all possible hyperplanes that might classify the data, SVM selects one with maximal distance (margin) to the nearest data points (support vectors).

When the classes are not linearly separable (there is no hyperplane that can split the two classes), a variant of SVM, called soft-margin SVM, chooses a hyperplane that splits the points as cleanly as possible, while still maximizing the distance to the nearest cleanly split examples. The method introduces slack variables, ξ_i, which measure the degree of misclassification of the datum x_i. Soft-margin SVM penalizes misclassification errors and employs a parameter (the soft-margin constant C) to control the cost of misclassification. Training a linear SVM classier solves the constrained optimization problem (2).

$$\min_{w,b,\xi_k} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \qquad s.t. \quad w \cdot x_i + b \geq 1 - \xi_i \tag{2}$$

In dual form the optimization problem can be represented by (3)

$$\min_{\alpha_i} \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j x_i \cdot x_j - \sum_{i=1}^{n}\alpha_i \qquad s.t. \quad 0 \leq \alpha_i \leq C, \quad \sum_{i=1}^{n}\alpha_i c_i = 0 \tag{3}$$

The resulting decision function $f(x) = w \cdot x + b$ has a weight vector $w = \sum_{k=1}^{n}\alpha_k y_k x_k$. Data points x_i for which $\alpha_i > 0$ are called support vectors, since they uniquely define the maximum margin hyperplane. Maximizing the margin allows one to minimize bounds on generalization error.

If every dot product is replaced by a non-linear kernel function, it transforms the feature space into a higher-dimensional one, thus though the classifier is a hyperplane in the high-dimensional feature space (see Figure 2). The resulting classifier fits the maximum-margin hyperplane in the transformed feature space. The kernel function can be defined as

$$k(x_i, x_j) = \Phi(x_i)\Phi(x_j) \tag{4}$$

where $\Phi(x)$ maps the vector *x* to some other Euclidean space. The dot product $x_i \times x_j$ in the formulae above is replaced by $k(x_i, x_j)$ so that the SVM optimization problem in its dual form can be redefined as: maximize (in α_i)

$$\tilde{L}(\alpha) = \sum_i \alpha_i - \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j k(x_i, x_j) \text{, s. t. } \sum_i \alpha_i y_i = 0; \quad \alpha_i \geq 0 \text{ for all } 1 \leq i \leq N \qquad (5)$$

A non-linear SVM is largely characterized by the choice of its kernel, and SVMs thus link the problems they are designed for with a large body of existing work on kernel-based methods. Some common kernels functions include:

- Linear kernel: k(x,x') = (x·x')
- Polynomial kernel: $k(x, x') = (sx \cdot x' + c)^d$
- RBF kernel: $k(x, x') = \exp(-\gamma(x - x')^2)$
- Sigmoid kernel: $k(x, x') = \tanh(s(x \cdot x') + c)$

Once the kernel is fixed, SVM classifiers have few user-chosen parameters. The best choice of kernel for a given problem is still a research issue. Because the size of the margin does not depend on the data dimension, SVM are robust with respect to data with high input dimension. However, SVM are sensitive to the presence of outliers, due to the regularization term for penalizing misclassification (which depends on the choice of C). The SVM algorithm requires $O(n^2)$ storage and $O(n^3)$ to learn.

The SVM method can also be applied to the case of regression. A version of SVM for regression, called support vector regression (SVR), was proposed by Drucker et al. [1997]. The basic idea of SVR is that a non-linear function learns by a linear learning method in a kernel-induced higher dimensional space. Similarly to how SVM classification ignores data points that are not support vectors, the SVR depend on a small subset of training data points.

The SVM's major advantage lies with their ability to map variables onto an extremely high feature space. This, in essence facilitates a means for the exploration of nonlinear kernel-based classifiers [Oladunni and Singhal, 2009; Burges, 1998], however, it has been discovered they do not favour large datasets, due to the demands imposed on virtual memory, and the training complexity resultant from the use of such a scaled collection of data [Horng et al., 2010]. Work from Fei et al. [2008] highlighted three "crucial problems" in the use of support vector machines. These are attaining the optimal input subset, correct kernel function, and the optimal parameters of the selected kernel, all of which are prime considerations within this study.

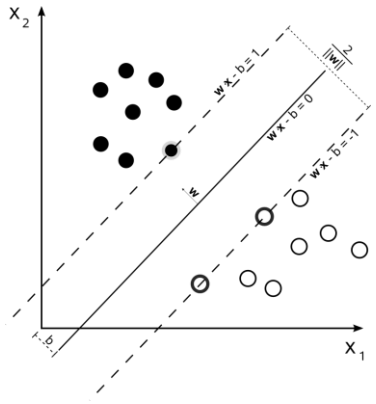


Fig. 1. Maximum-margin hyperplane for a SVM trained with samples from two classes. Samples on the margin are support vectors

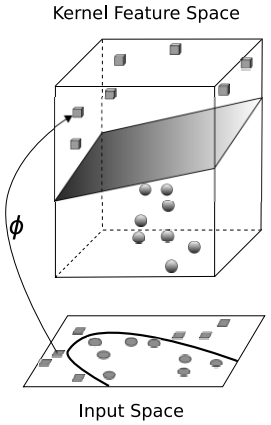


Fig. 2. Kernel function: a linearly inseparable input space can be mapped to a linearly separable higher-dimmensional space

Dataset and Preprocessing

This study uses a dataset that contains data from physical examination of patients, including mammographic and sonographic examinations, family history of breast cancer, and personal history of breast malignancy, all collected at Duke University Medical Centre [Jesneck et al., 2007]. Samples included in the dataset are those selected for biopsy only if the lesions corresponded to solid masses on sonograms and if both mammographic and sonographic images taken before the biopsy were available for review. Data contain 803 samples, 296 of which are malignant and 507 benign. Out of 39 descriptors, 13 are mammographic BI-RADS, 13 sonographic BI-RADS, 6 sonographic suggested by Stavros et al. [1995], 4 sonographic mass descriptors, and 3 patient history features. There is also a class label, -1 and 1, that indicates if a sample is malignant or benign.

The data features are as follows: mass size, parenchyma density, mass margin, mass shape, mass density, calcification number of particles, calcification distribution, calcification description, architectural distortion, associated findings, special cases (as defined by the BI-RADS lexicon [BI-RADS, 2003]: asymmetric tubular structure, intramammary lymph node, global asymmetry, and focal asymmetry), comparison with findings at prior examination, and change in mass size. The sonographic features are radial diameter, antiradial diameter, anteroposterior diameter, background tissue echo texture, mass shape, mass orientation, mass margin, lesion boundary, echo pattern, posterior acoustic features, calcifications within mass, special cases (as defined by the BI-RADS lexicon: clustered microcysts, complicated cysts, mass in or on skin, foreign body,

intramammary lymph node, and axillary lymph node), and vascularity. The six features suggested by Stavros [Stavros et al., 1995] are mass shape, mass margin, acoustic transmission, thin echo pseudocapsule, mass echogenicity, and calcifications. The four other sonographic mass descriptors are edge shadow, cystic component, and two mammographic BI-RADS descriptors applied to sonography—mass shape (oval and lobulated are separate descriptors) and mass margin (replaces sonographic descriptor angular with obscured). The three patient history features were family history, patient age, and indication for sonography [Jesneck et al., 2007].

Using the dataset in its original format for classification with SVM would be problematic due to the large amplitude of feature values caused by the different nature of the data variables and different units of measurements used. For example, the mass size values range from 0 to 75, as long as calcification ranges from 0 to 3. Such an inconsistency could affect the predictive abilities of a SVM classifier as some variables can be viewed as more 'influential' than others. The approach we used to address that problem was to process each data variable (data column) separately by transformation (6). It scales down the variables within the unit hypercube.

$$x_i^{new} = \frac{x_i^{old} - \min_i}{\max_i - \min_i} \tag{6}$$

We also explored how presence or absence of variables presented to the model for training and testing affects the classifier performance. Removing most irrelevant and redundant features from the data helps to alleviate the effect of the curse of dimensionality and to enhance the generalization capability of the model, and to speed up the learning process and to improve the model interpretability. The feature selection also helps to acquire better understanding about data and how they are related with each other. The exhaustive search approach that considers all possible subsets of variables is best for datasets with small cardinality, but impractical for large number of features as in our case. Jesneck et al. [2007] proposed a feature subset of 14 descriptors (s14) for their experiments with neural networks. They were derived by the stepwise feature selection technique. There is no guarantee, however, that an optimal variable selection for one classification technique will be optimal for another. In order to find the alternative selections for the SVM model we considered several feature selection algorithms, which generally fall into two categories: feature ranking and subset selection. The latter is more advanced and widely used in practice, which made us focus on it. We considered best first, subset size forward selection, race search, scatter search and genetic search combined with a set evaluation technique that considers individual predictive ability of each feature along with the degree of redundancy between them [Goldberg, 1989;

Hall, 1998]. We propose a set of 17 variables (s17) derived by the linear forward selection technique, proposed by Guetlien et al. [2009]. The feature set we obtained consists of the following variables: patient age, indication for sonography, mass margin, calcification number of particles, architectural distortion, anteroposterior diameter, mass shape, mass orientation, lesion boundary, special cases, mass shape, mass margin, thin echo pseudocapsule, mass echogenicity, edge shadow, cystic component, and mass margin. Two of these are general descriptors; three - mammographic BI-RADS; five - sonographic BI-RADS; four - Stavros'; and three - sonographic mass descriptors. The feature set is relatively balanced in representing different categories of data. In our experiments we also used the set of 14 variables (s14) mentioned above and the original full set of 39 variables (s39).

Empirical Results and Discussion

Using the training and testing datasets described above, we built a classification model based on SVM. For the purposes of the ROC we used the support vector regression technique, which outputs predictions as real numbers between -1 and 1, which mapped to the class labels (either -1 or 1). In order to minimize the bias in results associated with the random sampling of the training and testing data samples, we applied five-fold cross-validation, a.k.a. rotation estimation. The dataset was randomly spit into five mutually exclusive subsets (folds) of equal size. The model was trained and tested five times so that each time it was trained on one combination of four folds and tested on the remaining one. The cross-validation estimate of the overall model accuracy was calculated by (7).

$$CVA = \frac{1}{k}\sum_{i=1}^{k} A_i \quad , \tag{7}$$

where the number of folds k=5, CVA is the cross-validation accuracy, and A_i is the accuracy measure of the i-th fold (e.g. hit-rate, sensitivity, specificity).

The primary source for estimating the model accuracy is the confusion matrix (a.k.a. contingency table), illustrated in Figure 3. Results from experiments were summarized in four categories: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). The numbers along the primary diagonal in the matrix represent correct predictions, as long as those outside the diagonal represent the errors.

		True Class	
		Positive	Negative
Predicted Class	Positive	True Positive Count (TP)	False Positive Count (FP)
	Negative	False Negative Count (FN)	True Negative Count (TN)

Fig. 3. Confusion matrix for tabulation of classification results

In order to estimate the model performance we used the following derivations from the confusion matrix:

- True Positive Rate (TPR), a.k.a. sensitivity, hit rate, or recall is the ratio of correctly classified positives divided by the total positive count.

$$TPR = \frac{TP}{TP + FN} \tag{8}$$

- False positive rate (FPR), a.k.a. fall-out, or (1-specificity) is the ratio of incorrectly classified positives divided by the total negative count.

$$FPR = \frac{FP}{FP + TN} \tag{9}$$

- Accuracy (ACC) is the ratio of correctly classified instances (both positives and negatives) divided by the total number of instances.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

Estimating the accuracy of the built SVM model is important for the following two reasons: first, it can be used to estimate the future prediction accuracy, which could imply the level of confidence, the potential users may have; secondly, it can be used for choosing a particular instance of the SVM model among available options, e.g. selection of kernel function and parameter settings.

Accuracy is a common performance estimator in machine learning and data mining, but in many cases and problem domains it is not sufficient metric. Sometimes, accuracy can be

misleading, for example where important classes are underrepresented in the datasets and class distribution is skewed. In that case accuracy is helpless in counting different costs and consequences from misclassifications. This is the case of the domain we consider, as misdiagnosed malignant and benign samples have different consequences and even may cost life. Another drawback of the accuracy is that it depends on the classifier's operating threshold. When SVM runs as a regression function that outputs real numbers between -1 and 1, mapping outputs to class labels requires defining a threshold between -1 and 1, so that the output can fall below or above it, i.e. mapped to one or another class label. Applying different thresholds produces different instances of the classification model, each of which features a specific accuracy.

In order to address those accuracy deficiencies, we did Receiver Operating Characteristics (ROC) analysis [Fawcett, 2006]. This is a graphical assessment technique where the true positive rate is plotted on the Y-axis and false positive rate is plotted on the X-axis (Figure 4). In the ROC space, a classification model is a step curve plotted by connecting all model instances made by varying the threshold value.

The line that links (0,0) and (1,1) is the no-discrimination line. It represents the worst possible model, which predicts by a completely random guess. Any other classifier should appear above that line. If it pops below the line, a negation of its predictions would move it above the no-discrimination line.

On the other hand, the 'ideal' classifier would be represented by the point (0,1), the top-left corner, which shows that all true positives are found and no false positives are found. Any model performance can be measured by its proximity to the 'ideal' classifier. The closer the ROC curve is as a whole to the north-west corner, the better. That is also the most distant from the no-discrimination line, the better. Given a curve, the most 'north-west' point of the curve represents the model instance with maximal accuracy.

The ROC analysis also provides means for quantification of a model performance, these are Area Under the ROC Curve (AUC) and partial Area Under the ROC curve (pAUC) where sensitivity is above a certain value (p). The AUC / pAUC are scalars that measure the overall model performance, regardless of the operational threshold. The bigger the values, the better the model is. As long as AUC provides an overall estimation of the model, the pAUC is more relevant to the application area, as the potential users of CAD tools are particularly interested in working with high levels of sensitivity. It is believed that sensitivity above 90% ($_{0.90}$AUC) is relevant to the application field. Another clinically relevant metric that we estimated is specificity at given sensitivity. In order to be consistent with previous studies [Jesneck et al., 2007], we considered specificity at 98% sensitivity.

Table 1 summarizes results from numerous experiments where the SVM model was trained and tested using four kernels: linear, polynomial, RBF, and sigmoid. For each of those kernels we experimented with three different sets of variables: s39 that contains all

Table 1. Performance of SVM with linear, polynomial, RBF, and sigmoid kernels. Metrics for comparison include: area under the ROC curve (AUC), partial AUC at sensitivity above 90% ($_{0.90}$AUC), specificity at 98% sensitivity, and maximal accuracy (ACC_{max}). Models have been tested with three variable selections: s39, s17, and s14. Typical radiologist assessment values are also included. Figures in bold show best values.

SVM linear	*s39*	*s17*	*s14*	*Radiologist*
AUC	**0.91**	**0.91**	0.89	0.92
$_{0.90}$AUC	0.71	0.74	0.62	0.52
Spec /98% sens	**0.36**	0.30	0.22	0.52
ACC_{max}	0.84	**0.85**	**0.85**	n/a

SVM polynomial	*s39*	*s17*	*s14*	*Radiologist*
AUC	**0.91**	**0.91**	0.89	0.92
$_{0.90}$AUC	0.72	0.74	0.62	0.52
Spec /98% sens	**0.36**	0.29	0.23	0.52
ACC_{max}	0.84	**0.85**	0.84	n/a

SVM FBF	*s39*	*s17*	*s14*	*Radiologist*
AUC	0.90	**0.91**	0.88	0.92
$_{0.90}$AUC	0.64	**0.75**	0.58	0.52
Spec /98% sens	0.29	0.32	0.20	0.52
ACC_{max}	0.83	**0.85**	0.83	n/a

SVM sigmoid	*s39*	*s17*	*s14*	*Radiologist*
AUC	**0.91**	**0.91**	0.88	0.92
$_{0.90}$AUC	0.67	**0.75**	0.62	0.52
Spec /98% sens	0.27	**0.36**	0.20	0.52
ACC_{max}	**0.85**	0.84	0.83	n/a

The table figures show that the selection of descriptors for training and testing plays a significant role in the SVM performance. According to all metrics and no matter which kernel is selected, it is evident that the variable set s14, proposed by Jesneck et al. [2007] for classification with MLP is outperformed by both s39 and s17. As mentioned before, that is not surprising as an optimal variable selection for one classification model would not be optimal for another. We also show that the alternative selection of variables, s17, can outperform both s14 and s39. That selection significantly improves the $_{0.90}$AUC of s14 from 12% to 17%, depending on which kernel is used, and also outperforms the radiologist value by 23%. The other clinically relevant metric, specificity at 89% sensitivity, is also improved by s17 in comparison with s14 - from 6% to 16%. In some cases s39 performs as well as s17, but it never gets better. The variable set s17 shows itself as the best performer regarding AUC and ACC_{max} with only few exceptions.

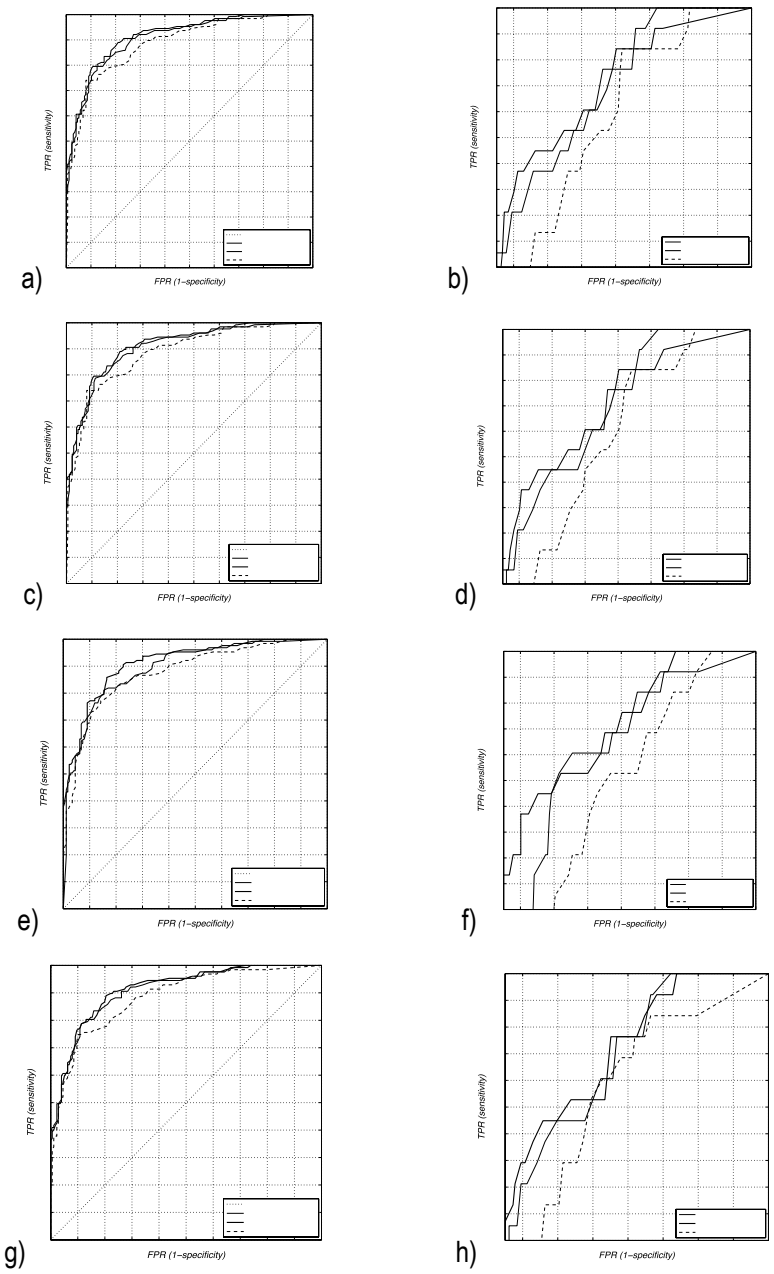


Fig. 4. Performance of SVM with linear, polynimial, RBF, and sigmoid kernels with three variable sets: all attributes (s39); selection of 17 attributes based on the subset size forward selection method (s17) Guetlin et al. [2009]; and selection of 14 attributes proposed by Jesneck et al. [2007]

We also explored how choice of the kernel function influences the SVM predictive abilities. This is particularly important when data belong to classes, which are not linearly separable. In those cases we can expect that the SVM model with linear kernel wouldn't perform well in contrast to the non-linear ones. Table 1 shows the results for each kernel function. The SVM works well with all of them. Considering s17 only, AUC for all four kernels is 91% and $_{0.90}$AUC is from 74% to 75%. Specificity at high sensitivity, however, varies with different kernels. Regarding this metric, the sigmoid kernel outperforms the others, followed by the RBF, linear, and polynomial.

Our findings also could be compared with those from studies that use models based on the most common neural networks - MLPs, given that all methods use the same dataset and variable sets (Nachev & Stoyanov, 2010). SVM shows the same AUC as MLP, but improves $_{0.90}$AUC by 7% (68% vs. 75%) and max accuracy by 2% (83% vs. 85%).

Fig. 4 gives further details on the SVM ROC analysis, The left-hand figures illustrate the ROC curves and AUC for each kernel and variable set; the right-hand figures illustrate the area of sensitivity above 90% and list $_{0.90}$AUC.

Conclusion

This study explores support vector machines utilized as predictors of malignant breast masses, trained and tested with data from mammographic and sonographic examinations. We used data collected from Duke University Medical Centre, which contains 39 descriptors. Our study was focused on two issues: how reduction of dimensionality of the training and testing data affect the discriminatory power of the model; and how choice of the SVM kernel function and model parameters affect its predictive abilities.

In order to quantify the model performance we did ROC analysis and utilized metrics, such as true positive rate, false positive rate, area under the ROC curve, partial area under the ROC curve, and specificity at high sensitivity.

Our results show that the reduction of dimensionality plays a significant role in the model performance. We propose a set of 17 variables, which outperforms the 14 variables set of Jesneck et al. [2007]. The choice kernel function among linear, polynomial, RBF, and sigmoid, however, does not influences the model performance, with exception of one metric - specificity at high sensitivity. In that case the sigmoid kernel is the best performer. The fact that the linear kernel shows similar performance to that of the non-linear kernels is an indication that the feature space is linearly separable and data points are distributed in a way that makes the classification task linear in terms of complexity.

We also found experimentally that the SVM outperform a common classification technique used in the field - MLP neural networks. SVM shows the same AUC, but improves $_{0.90}$AUC by 7% (68% vs. 75%) and max accuracy by 2% (83% vs. 85%).

In conclusion, we believe that SVM is a promising technique for breast cancer diagnosis, but when used, it requires a careful reduction of dimensionality and well-selected model parameters.

Bibliography

[BI-RADS, 2003] American College of Radiology. BI-RADS: ultrasound, 1st ed. In: Breast imaging reporting and data system: BI-RADS atlas, 4th ed. Reston, VA: American College of Radiology, 2003

[Burges, 1998] Burges, C. A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery, 2, 121-167, 1998.

[Chen et al., 2009] Chen, S., Hsiao, Y., Huang, Y., Kuo, S., Tseng, H., Wu, H., Chen, D.: Comparative Analysis of Logistic Regression, Support Vector Machine and Artificial Neural Network for the Differential Diagnosis of Benign and Malignant Solid Breast Tumors by the Use of Three-Dimensional Power Doppler Imaging. Korean J Radiol vol. 10, 464-471 2009.

[Drucker et al., 1997] Drucker, H. Burges, C., Kaufman, L., Smola, A., and Vapnik, V., Support vector regression machines, Advances in Neural Information Processing Systems 9, pages 155-161, Cambridge, MA, MIT Press, 1997.

[Fawcett, 2006] Fawcett, T. “An introduction to ROC analysis”; Pattern Recognition Letters, Vol. 27 Issue 8, pp. 861-874, 2006.

[Fei et al., 2008] Fei, L., Li, W. & Yong, H. Application of least squares support vector machines for discrimination of red wine using visible and near infrared spectroscopy. Intelligent System and Knowledge Engineering, ISKE' 08, 2008.

[Goldberg, 1989] Goldberg, D.: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley: Reading, MA, 1989.

[Guetlein et al., 2009] Guetlein, M., Frank, E., Hall, M., Karwath, A. “Large Scale Attribute Selection Using Wrappers”; In Proc. IEEE Symposium on CIDM, pp.332-339, 2009.

[Hall et al., 2009] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten. I., 2009. The WEKA data mining software: an update. SIGKDD Explor. Newsl. 11, 1, 10-18, 2009.

[Horng et al., 2010] Horng, S., Su, M., Chen, Y., Kao, T., Chen, R., Lai, J. and Perkasa, C. A novel intrusion detection system based on hierarchical clustering and support vector machines. Expert Systems with Applications, 38, 306-313, 2010.

[Jesneck et al., 2007] Jesneck, J., Lo, J., Baker, J. “Breast Mass Lesions: Computer-Aided Diagnosis Models with Mamographic and Sonographic Descriptors”; Radiology, vol.244, Issue 2, pp 390-398, 2007.

[Kopans, 1992] Kopans D. “Standardized mammographic reporting”; Radiol Clin North Am, Vol. 30, pp. 257–261, 1992

[Lacey et al., 2002] Lacey, J., Devesa, S., Brinton, L. “Recent Trends in Breast Cancer Incidence and Mortality.”; Environmental and Molecular Mutagenesis, Vol. 39, pp. 82–88, 2002.

[Nachev & Stoyanov, 2010] Nachev, A. and Stoyanov, B., “An Approach to Computer Aided Diagnosis by Multi-Layer Preceptrons”, In Proceedings of International Conference Artificial Intelligence (IC-AI’10), Las Vegas, 2010.

[Oladunni and Singhal, 2009] Oladunni, O. O. & Singhal, G. 2009. Piecewise multi-classification support vector machines. International Joint Conference on Neural Networks, IJCNN'09, 2009.

[Stavros et al., 1995] Stavros, A., Thickman, D., Rapp, C., Dennis, M., Parker, S., Sisney, G. "Solid Breast Modules: Use of Sonography to Destinguish between Benign and Malignant Lesions"; Radiology, Vol. 196, pp. 123-134, 1995.

[Vapnik, 1995] Vapnik, V., The Nature of Statistical Learning Theory. Springer, New York (1995)

Authors' Information

Anatoli Nachev *– Business Information Systems, Cairnes Business School, National University of Ireland, Galway, Ireland; e-mail: anatoli.nachev@nuigalway.ie*

Major Fields of Scientific Research: data mining, neural networks, support vector machines, adaptive resonance theory.

Mairead Hogan *– Business Information Systems, Cairnes Business School, National University of Ireland, Galway, Ireland; e-mail: mairead.hogan@nuigalway.ie*

Major Fields of Scientific Research: HCI, usability and accessibility in information systems, data mining.

DECISION MAKING SUPPORT AND EXPERT SYSTEMS

UTILITY FUNCTION DESIGN ON THE BASE OF THE PAIRED COMPARISON MATRIX*

Stanislav Mikoni

***Abstract**: In the multi-attribute utility theory the utility functions are usually constructed by dots. It concerns both the lottery's method and the value increasing method. In the both cases the utility function is constructed in the absolute scale [0,1] that causes inconveniences for experts. The comparative assessments look more preferable for decision-makers. The paired comparison matrix (PCM) looks as a natural model representing the preference structure of decision-maker (DM).*

We use scale points of attributes as a PCM comparative entities. We use also increasing/decreasing entity priority as a criterion. Function of priorities is transformed to utility function on the base of a normalizing function. Such a function allows using the matrix power as parameter affecting the form of utility function.

The PCM provides the extended possibilities to DMs to form comparative assessments both the qualitative ones (as better-worse) and the quantitative ones reflecting winnings and losses of DMs. In the paper we consider methods for utility function construction having different forms of its presentation. Among them there are utility functions based on attributes measured in nominal scales.

***Keywords**: utility function, paired comparison matrix, scale points, priority function.*

***ACM Classification Keywords**: H.4.2 [Information Systems Applications]: Types of Systems-decision support*

Introduction

The multi attribute utility theory demands a utility functions construction for each attribute. Two well known methods are used for this goal. First of them is lottery method proposed

* The work had been fulfilled under financial support of Russian Fundamental Research Fund (project № 10-01-00439)

by von Neumann and Morgenstern. The second of them is based on value estimation by expert for some scale points. Both of the methods have such disadvantage as necessity of absolute quantity values assessment. According T. Saaty, relative values more convenient for expert than absolute ones [Saaty, 1996]. These values are used under paired comparison matrix forming as preference relation. Thus paired comparison matrix (PCM) contains an expert preference structure. We will use scale points of attributes as an PCM comparative alternatives. In the paper we will consider the problem of utility functions construction on paired comparison matrix base.

Preference representation on scale points

Let Z be scale points set. Then preference relation R on a set Z is subset on the product $Z \times Z$: $R \subset Z \times Z$. When cardinality of Z is small, the preference relation may be conveniently represented by the $n \times n$ matrix **A**. Its element a_{ij}, $\forall i$, $j \in \{1,\ldots,n\}$, is interpreted as the preference degree of the scale points z_i over z_j.

We will consider three kinds of preference relation: binary relation with $a_{ij} \in \{0, 1\}$, probabilities relation with $a_{ij} \in [0, 1]$, multiplicative or ratio relation with $a_{ij} \in [1/N, N]$, where $a_{ij}=N$ denotes that attribute useful in point z_i is N times as good as in point z_j.

These relations are represented by reciprocal matrices. For binary and probabilities matrices a_{ij}, $\forall i$, $j \in \{1,\ldots,n\}$, are calculated as $a_{ji}=1 - a_{ij}$. For multiplicative preference matrix $a_{ji} = 1/a_{ij}$. To construct reciprocal matrix it is need to make $n(n-1)/2$ comparisons between alternatives.

Beside reciprocal matrices we will use non-reciprocal matrices in which $a_{ji} \neq f(a_{ij})$. We will name the kind of preference represented by a such matrix as benefit / losses. For example, if footballs team A have wined team B with 3:1 score, matrix element $a_{ij}=3$ is interpreted as the benefit of team A and $a_{ji}=1$ is interpreted as the loss of that team. To construct non-reciprocal matrix it is need to make $n^2 - n$ comparisons between alternatives.

Hence there are some ways to construct consistent matrix from the set of n or $n - 1$ comparison. To construct non-reciprocal matrix it is enough to assess one from its rows or columns. Using known values of the first column a rows of the matrix are formed. All cells of the row accept the value from the first cell. If a values of the first column are rising from up to down, then $a_{ji} > a_{ij}$, $i \neq j$, $\forall i$, $j \in \{1,\ldots,n\}$. Another words all elements of the bottom triangle sub matrix are bigger then a corresponding elements of the upper triangle sub matrix. Analogous the matrix is constructed on the base of the known first row. That matrix

contains the opposite preferences, because all elements of the upper triangle sub matrix are bigger then a corresponding elements of the bottom triangle sub matrix.

Another way of a matrix construction is to assess the cells of a matrix corresponding Hamilton path on their graph. The procedure demands only n –1 values of preference relation. That values are entered into the cells of matrix parallel their main diagonal. On the next step a values are calculated for remaining cells of a matrix. There are developed a methods of single-digit finding of values for remaining cells of a matrix [Alonso S. et al.]. In paper [Kiselev 1, 2011] the task is solved as optimization one with new consistency criterion.

When all cells of paired comparison matrix have been assessed we can calculate the dominance degree between scale points. A paired comparison matrix only captures the dominance of one scale point over each other points in one step. The dominance is accumulated by raising the matrix to the next power beginning the first one. Tomas Saaty had proved how to obtain a relative scale among n alternatives from their paired comparison matrix. The relative dominance of an alternative is given by the solution of the eigenvalue problem $\mathbf{Aw}=\lambda_{max}\mathbf{w}$. Normalized eigenvector corresponding matrix eigenvalue $=\lambda_{max}$ represents finite dominance vector $\mathbf{w}=(w_1,\ldots,w_j,\ldots,w_n)$ of the alternatives, where

$$\sum_{j=1}^{n} w_j = 1 \qquad (1)$$

In fact dominance a vector w represents a discrete priority function determined on the scale points. To receive a discrete utility function u from a priority function w the last one transforms by $(w_i - w_{\min})/(w_{i,\max} - w_{\min})$. Values of utility function are belonged to interval [0, 1]. It should be noted that the linear dependence exists between utility function (UF) and priority function (PF). So it is enough to investigate the only priority function properties.

Linear utility functions construction

To create linear priority function it is necessary to maintenance even change of dominance of the i-th scale point over j-th point, $\forall i, j \in \{1,\ldots,n\}$, $i \neq j$. Dominance change magnitude depend on a kinds of preference and a matrix content. Even change of dominance is satisfied by a binary matrix which consist of the triangle sub matrix with cells $a_{ji}=1$ and another triangle sub matrix with cells $a_{ij}=0$. In such matrix the dominance difference between neighboring points equal 1. In the Table 1 the example of matrix 6×6 is shown.

The binary matrix is placed on the left side of the table 1. Its diagonal elements $a_{ii}=1$, $\forall i, j \in 1,\ldots,n\}$ to receive the smallest priority $w_{i,\min}>0$. The numbers in the column "Score" are the sums of ones in the corresponding matrix rows. Priority function values are calculated by score numbers normalization. In the column "Useful" a values calculated on base of the priority function are placed.

Table 1

Scale points	1	2	3	4	5	6	Score	Priority	Useful
1	1	0	0	0	0	0	1	0,0476	0,1665
2	1	1	0	0	0	0	2	0,0952	0,3331
3	1	1	1	0	0	0	3	0,1428	0,4995
4	1	1	1	1	0	0	4	0,1904	0,6660
5	1	1	1	1	1	0	5	0,2380	0,8325
6	1	1	1	1	1	1	6	0,2859	1,0000

The linear graphic of the utility function is shown in Fig. 1.

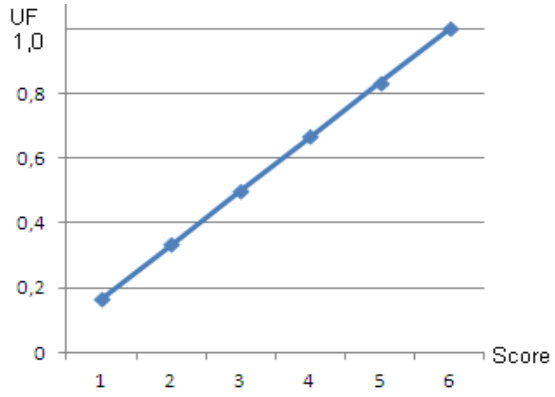


Fig. 1. The linear graphic of the utility function.

The discrete function points are connected to see the function form. The utility function with descending form is created on base of binary matrix which consist of the upper triangle sub matrix with cells $a_{ji}=1$ and bottom triangle sub matrix with cells $a_{ij}=0$.

A linear utility function can be created on the base matrix "benefit / losses" too. An example of such matrix is shown in Table 2.

Table 2.

Scale points	1	2	3	4	5	6	Score	Priority	Useful
1	1	1	1	1	1	1	6	0,05405	0,19353
2	2	1	2	2	2	2	11	0,09910	0,35484
3	3	3	1	3	3	3	16	0,14414	0,51611
4	4	4	4	1	4	4	21	0,18919	0,67742
5	5	5	5	5	1	5	26	0,23423	0,83869
6	6	6	6	6	6	1	31	0,27928	1,00000

The cells of each matrix row beside diagonal cells have the same values equal to cell values of the first column. The column sell values (1, 2, 3, 4, 5, 6) are obtained with a such generating function as arithmetic progression with step 1.

Non-linear utility functions construction

To create non-linear priority function it is necessary to maintenance variable change of dominance of the *i*-th scale point over *j*-th point, $\forall i, j \in \{1,\ldots,n\}$, $i \neq j$. This objective can be achieved by three ways: corresponding generating function choice, transforming one preference kind to another, change of parameter of priority function calculation on base PCM. This parameter is power of raising *k* the matrix **A**.

The simplest monotonic generating functions are geometric progression with constant step and Fibonacci function. An example of geometric progression is 2^i function, $i \in \{1,\ldots,n\}$. The function generates number consequence: 1, 2, 4, 8, 16,… The Fibonacci function generates number consequence: 1, 1, 2, 3, 5, 8,… The difference change between neighboring number consequence magnitudes determines the velocity of increasing or decreasing function. An example of non-monotonic generating function is Newton binomial one. A generating function can be applied for assessment of the first column or row the matrix and Hamilton path cells of corresponding graph.

In Fig. 2 the example is shown of non-linear priority function construction by transforming preference “benefit / losses” preference kind a_{ij}^{bl} (see matrix in Table 2) to probabilities preference kind a_{ij}^{pr} according formula:

$$a_{ij}^{pr} = \frac{a_{ij}^{bl}}{a_{ij}^{bl} + a_{ji}^{bl}} \qquad (2)$$

Two curved shown in Fig. 2 characterize non-inclination decision maker to risk.

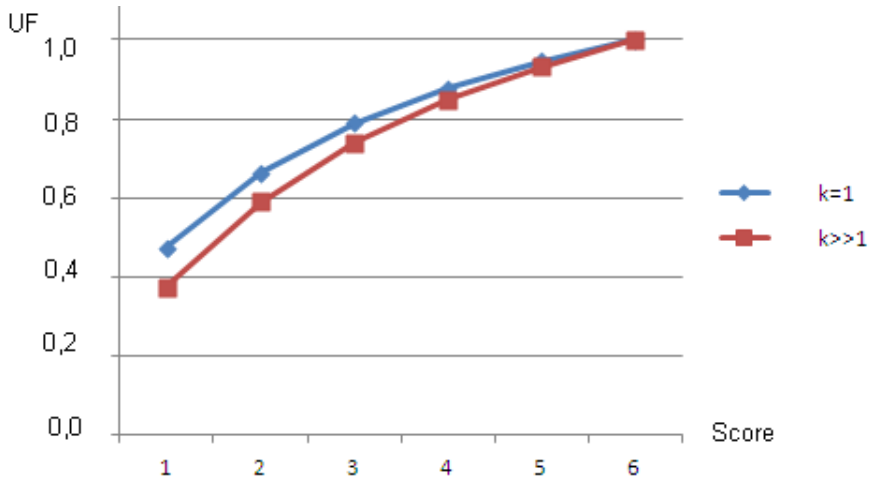


Fig. 2.

The upper utility function calculated by raising the matrix to first power (k=1) and bottom utility function calculated on base eigenvector of matrix by raising the matrix to power k>>1. Thus the example demonstrates both way of maintenance variable change of dominance of the i-th scale point over j-th point, transforming one preference kind to another and change of parameters of priority function calculation on base PCM.

Utility functions construction under known PCM

In multi criteria tasks beside quantitative criteria are often applied qualitative criteria too. For example, quality work can be characterized by manufacturing firm. To calculate value of multi attributes useful function it is necessary to transform firm names into numerical estimations. It can be made with expert help. Hence more objective estimations can be received if it is known results of firm interaction. Let firm interaction be meant patents trading and each firm is interested to sell more patents then to buy them. That firm interaction is represented by “benefit / losses” matrix shown in Table 3.

Under PCM the graphic of the utility function (UF) calculated on the base its eigenvector (matrix power k>>1) is shown in Fig. 3. To facilitate analyses the function values are replaced under corresponding matrix columns (firm names). The matrix shown in Fig. 3

has a bad consistency. It is confirmed by circles passing through corresponding graph vertexes. Number of circles passing through each vertex is shown in table 3 last row. The total number of circles into graph is equal 12 and maximum number is equal 20.

Table 3

Firm name	1	2	3	4	5	6	Score	UF, k=1	UF, k>>1
1	1	3	2	4	1	2	13	1,0000	0,9686
2	1	1	2	1	2	1	8	0,6154	0,6011
3	4	2	1	3	1	2	13	1,0000	1,0000
4	3	1	4	1	2	1	12	0,9231	0,9480
5	1	3	2	2	1	1	10	0,7692	0,7266
6	3	1	1	3	2	1	11	0,8462	0,8508
Cycles	5	7	7	6	5	6			

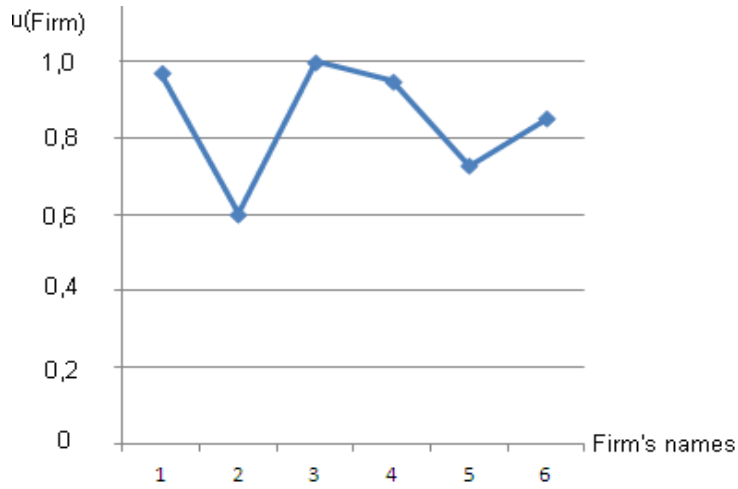


Fig. 3.

A biggest percent of circles into the graph indicates on a bad consistency of PCM. Hence consistency concept don't applied to matrices presented a competition results. To evaluate competitor's aggression level Igor Kiselev had proposed total preference factor [Kiselev 2, 2011].

The priority functions perturbation under increasing the matrix power is well seen in Fig. 4. The matrix power corresponds abscises axis represented in logarithmic scale. The vertical line in the Fig. 4 indicates matrix power k=1 or ln k = 0. On the left side from point ln k = 0 priority functions of alternatives are aspirated to 1/N that corresponds to matrix power

k = 0. On the right side from point ln k = 0 priority functions of alternatives are aspirated to the eigenvector. The eigenvector values are marked on right vertical line. In that matrix power point the perturbation process is over. It was reason why the UF with k>>1 had been choose in Fig. 3.

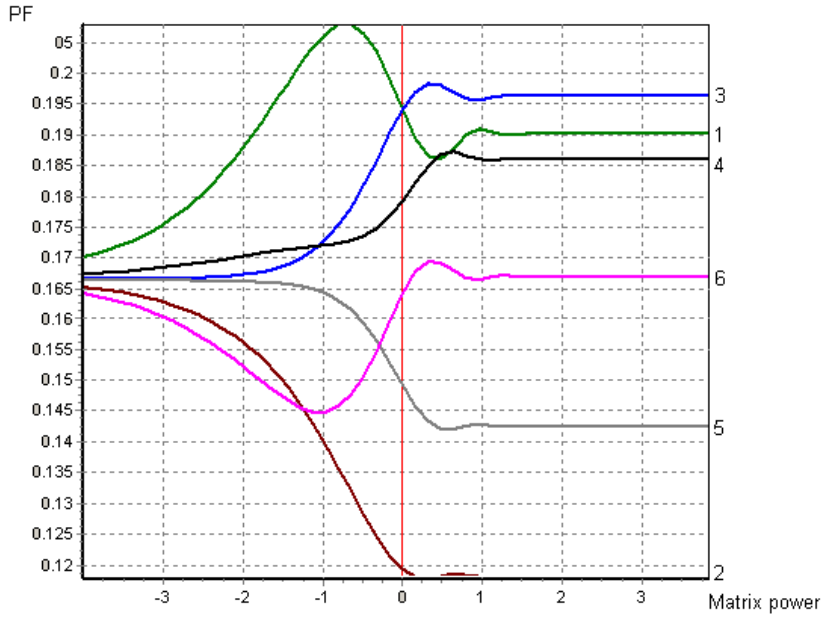


Fig. 4.

Conclusion

In addition to well-known methods of utility functions construction the new method is proposed based on paired comparison matrix using. We use scale points of attributes as a PCM comparative entities. The main problem of such approach is to assess the matrix content. Two ways of a problem solving are applied. One from them is expert method and another way is to tournament matrix using. To facilitate expert assessment of matrix content the shorten way is proposed. That way is based on generating function using. With a generating function help only n or n – 1 sells is assessed. The remaining cells are assessed automatically under consistency factor ensuring. The decision maker must only choice preference kind of matrix and generating function type.

Tournament matrices are used when results of active entities interaction are known. They are applied for transforming of nominal values to numerical ones, for instance to assess firms importance. Such assessments permit us to use qualities' attributes under multi-attribute values computing.

The convention and effectively of proposed method was confirmed by numerous experiments on PCM applying for utility functions construction. The all experiments had been fulfilled on the choice and ranking system "SVIR-R". The system had been elaborated in St. Petersburg State Transport University under author direction [www.mcd-svir.ru].

Bibliography

[Saaty, 1996] Saaty T.L. The Analytic Network Process: Decision Making With Dependence and Feedback. RWS Publications, 1996. P. 370.

[Alonso S. et al.] A consistency based procedure to estimate missing pair wise preference values // International Journal of Intelligent Systems. 2009. Vol. 23, № 2. P. 155–175.

[Kiselev 1, 2011] Index of quantitative preference consistency in pair wise comparison matrix // Bulletin of the Tomsk Polytechnic University. 2011, Vol.318, № 5, P. 22–24.

[Kiselev 2, 2011] Analysis models and algorithms of different expert preferences types based on the paired comparison matrices. The doctors thesis // St. Petersburg, PGUPS, 2011.

Authors' Information

Stanislav Mikoni *– professor of St. Petersburg State Transport University, St. Petersburg 190031, Russia;*
e-mail: svm@sm4265.spb.edu

Major Fields of Scientific Research: System analyses, Multicriteria choice, Intelligent technologies, Simulation.

PRINCIPLES OF THE DEVELOPMENT OF INTERACTIVE QUERY EXPERT SYSTEMS

Valentin Kataev

***Abstract**: This paper describes the principles for the development of interactive query or question-answer expert systems (ES) in the Multi Studio software environment in Multi (the universal language), as well as in such environments as Visual Prolog and CLIPS. The paper also presents a comparative analysis of those principles checked by solving the test problems to show a significant advantage of Multi in the development of those ES according to the main quality parameters: language usability, work content of ES development, electronic memory size and speed of the developed ES.*

As a result of our study we formulate several principles of developing ES tool environments:

- *Development of a super-high-level universal environment language by integrating the best qualities of all the languages considered in the paper.*
- *Development of a universal structure of a knowledge base with a unitized syntax based on semantic networks.*
- *Development of a hybrid tool environment which can separately perform the following:*
  - *A one-time translation (compilation) of program and data input texts into an internal language of a knowledge base*
  - *Multiple fetch of programs from a knowledge base (the programs are executed by interpretation of those programs' instructions in a hybrid environment)*

***Keywords**: expert systems, CLIPS, Multi Studio, Prolog.*

***ACM Classification Keywords**: I.2.5 Programming Language and Software.*

Introduction

Being computer software QAES (ES), on the one hand, belong to data retrieval systems (ACM Computing Classification System (CCS) code: H.3.4) and, on the other hand, ES belong to decision-making systems (CCS: I.1.3).

Such systems are designed to send inquiries to a user online to receive certain answers from that person; as the required minimum of the answers is collected, the system generates special recommendations for the user what he or she is supposed to know or to execute.

The suggested recommendations (solutions), in general, can be informative, advising and also controlling. In the last case the question-answer expert system is built in a program-technical system as a control computing unit, when the information is input into the ES through a control equipment and then the ES outputs the generated controlling information to operating mechanisms.

The considered systems can be successfully applied in any range of human activity. A modern personal computer is quite a sufficient device to install and use rather advanced complex expert systems. But reality is far not so good.

The main reason is *complexity* of application development tools and maintenance of ES.

The present paper considers this issue by example of three toolsets applicable for development of information and advising systems including CLIPS, Multi Studio, and Visual Prolog.

Tools for ES Development

A standard software technology for developing ES is shown in Fig. 1.

Any ES is computer software. To develop it one always needs some other software (development environment) which converts algorithms of a certain developed ES into machine codes. There are two types of environments: compilers and interpreters.

The first ones develop stand-alone systems which can be run independently from their development environment. The second ones interpret input (source) language instructions i.e. they play the role of ES. The interpreters here can be subdivided into “pure” (i) and “hybrid” (ci). First the ci-environment translates (compiles) a text from the input (source) language into the internal language of the environment and saves the resulting structures in a knowledge base. This procedure is followed by a multiple execution (interpretive translation) of the environment internal instructions (i.e. ES functions are performed). Chart 1 demonstrates all of three versions of ES development.

Stand-alone ES(c):

- A program is securely built in the system
- Higher speed (despite for ES it does not matter as the most of computer time is “consumed” by a user).
- Complicated maintenance (every modification requires recompilation of programs).

- A compiling environment is, as a rule, universal with a universal classical language of a not very high level which requires a programmer of high qualification).

Integrated ES(i):

- A specially designed environment with a specialized language which can be of higher levels than that one of a compiling environment.
- Can execute an infinite number programs.
- Stand-alone data storage of input texts in the environment language in personal files of each program.
- Input texts are analyzed every time the ES is started.

Integrated ES(ci):

- As opposed to the environment mentioned above, the hybrid CI-environment first translates (compiles) the input (source) text into the internal language of the environment and saves the obtained structures into the knowledge base. Then this procedure is followed by a multiple execution (interpretive translation) of the environment internal instructions (i.e. ES functions are performed).
- The knowledge base is meant to store lots of programs and all the necessary data for them.
- The input language is preferably to be the universal higher-level language with a subset of instructions for effective creation and maintenance of ES. Otherwise, the use of the powerful system may appear to be improvident, wasteful and limited.
- Lower requirements of programmer skills or programming of ES can be performed by a knowledge engineer.

All other conditions being equal, we consider that the best variant is the hybrid environment. Below we give characteristics of three development systems.

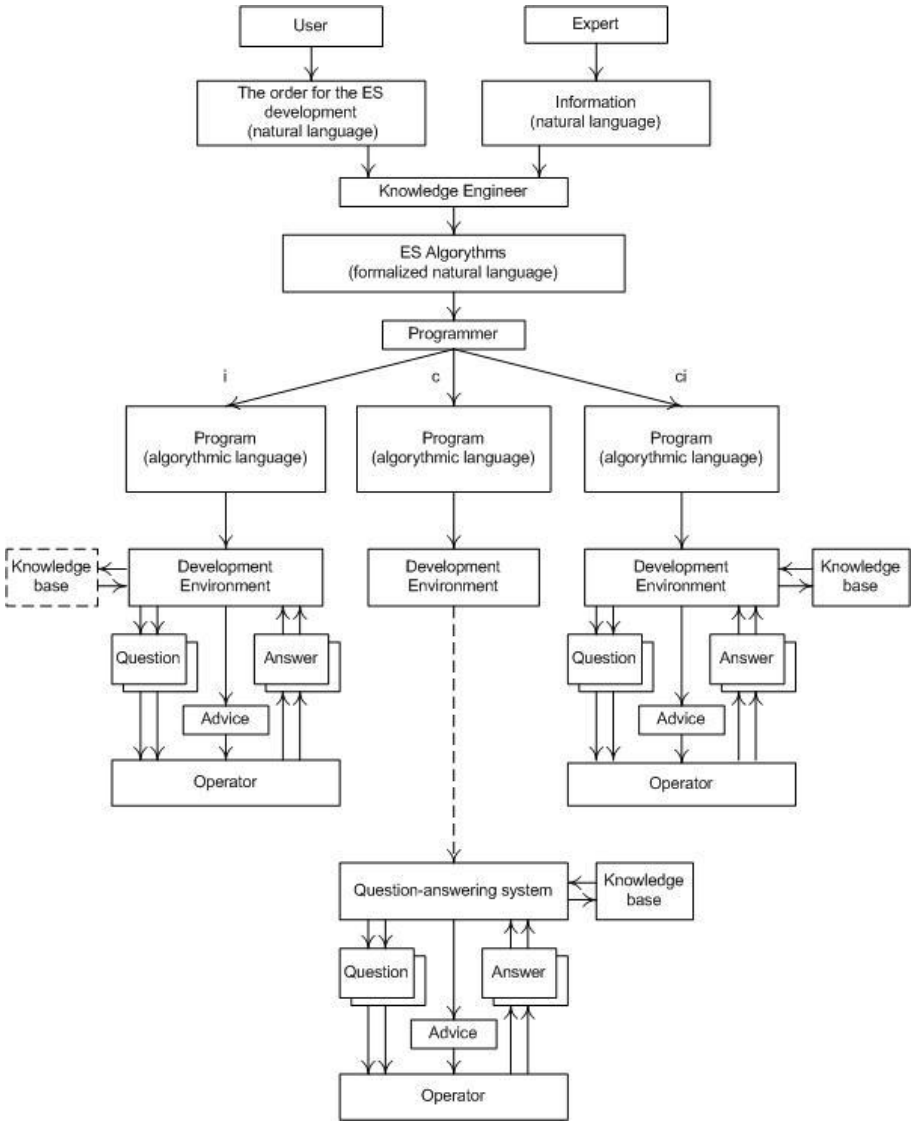


Fig. 1. Versions of ES development

CLIPS Environment

CLIPS is a pure interpreter specially designed for developing expert systems.

Its basic developer is NASA (USA). The first version dates back to 1984. The last version is available since 6.30 2008.

Global language syntax: nested LISP-lists. Example: (X (+ 25 3)).

Data: facts (ordered and unordered). The last ones are represented by frames which consist of slots containing fact names and fact values. Example: Name (Bobby), year of birth (1997). The ordered facts do not contain names. In the list of facts they are ordered strictly by index. Variables can be of any type.

The program language is C-like.

Programs consist of modules which consist of rules and functions. The rules are arranged in a module in random order (sequence). The data exchange between modules is performed by functions "export" and "import", between the rules – by global variables. The rule basic structure: the module name, rule name, priority, formula. The rule formula: left side => right side.

The left side contains the conditions under which the program starts to execute the right side of the formula. Those conditions may include parameters for fact search and/or references to the rules which must precede the execution of this rule in the program stack. The right side of the rule may contain instructions and functions that implement the various actions. In particular, there may be queries to the PC operator, displayed in a dialog box, response message handlers and the output of the resulting recommendations for the operator to see. Also there may be the instruction of exit from the module (program).

A knowledge base is a set of text files containing data and programs in the source language. Each program must have its personal set of files.

Execution of a program includes three steps:

- CLIPS RAM emptying.
- Downloading text from files to RAM and dynamic forming of fact and rule lists.
- The program start and its execution (interpretation of the rules).

The program execution Technique.

The program performs sequential execution of the modules, beginning with the startup (MAIN). Inside the module, the start rule is the first to execute (this rule is automatically generated by the program itself). Then all the rules of the module are cyclically searched through in order to check the conditions to determine the feasibility of their execution. The selected rules are put in the operational queue in descending order of priority. If there are more than one rule of higher priority, then the program selects among them an only one according to the current conflict resolution strategy selected by the user (out of 7 possible strategies). The rule selected in such a way is to be activated and executed.

Then the process of selecting a new rule to be executed repeats. The module (program) execution is terminated if the next rule in turn ends with the "return" instruction or a new operational turn is empty.

An interface of the CLIPS system is very modest. A standard dialogue with the operator is performed in the main black-and-white window of the system in a command line mode.

Logging of ES execution is also performed in the main window.

Text coloring in windows (software or manual) is not supported. The system works with texts in Latin only, at least Version 6.30 (03/26/08).

As an example we take a small expert system for searching malfunctions in an automobile engine (AUTO) which can be found on a CD enclosed to [Джарратано, 2007].

As the program is pretty big, in Listing 2 we demonstrate just two rules with comments, and Listing 3 presents algorithms of this ES.

Listing 2. Two rules of AUTO program written in CLIPS.

```
(defrule determine-rotation-state "" ; rule 1
   (working-state engine does-not-start) ; rule gets activated if the engine does not start
   (not (rotation-state engine ?)) ; engine shaft did not rotate
   (not (repair ?)) ; no solution (recommendation)
   =>
   (if (yes-or-no-p "Does the engine rotate (yes/no)? ") ; does the engine rotate?
      then
         (assert (rotation-state engine rotates)) ; if YES then the engine shaft rotates
         (assert (spark-state engine irregular-spark)) ; at that, the spark is irregular
      else
         (assert (rotation-state engine does-not-rotate)) ; if NO then the shaft does not rotate
         (assert (spark-state engine does-not-spark))) ; at that, the engine does not spark )
(defrule determine-battery-state "" ; rule 2
```

(rotation-state engine does-not-rotate) *; rule gets activated if the engine shaft does not rotate*

(not (charge-state battery ?)) *; battery is not charged*

(not (repair ?)) ; *no solution*

=>

(if (yes-or-no-p "Is the battery charged (yes/no)? ") *; is the battery charged?*

then

(assert (charge-state battery charged)) *; YES, the battery is charged*

else

(assert (repair "Charge the battery.")) ; *NO, solution: Charge the battery*

(assert (charge-state battery dead))) *; the battery is dead*

)

Listing 3. A Fragment of logging of ES in CLIPS.

Does the engine rotate (yes/no)? n

Is the battery charged (yes/no)? n

Suggested Repair:

Charge the battery.

CLIPS>

Visual Prolog

Visual Prolog is an interpreter [Адаменко и др., 2003].

Basic developer: Prolog Development Center (Denmark), since 1984. (Turbo Prolog => Prolog PDC => Visual Prolog).

Global syntax: the list of sentences.

The Prolog program includes the following sections: DOMAINS, PREDICATES, CLAUSES, GOAL GOAL contains a list of target predicates. The section named PREDICATES contains predicate declarations (names and types of arguments), DOMAINS specifies nonstandard (custom) types of predicate arguments (predicate variables). CLAUSES contains detailed descriptions of facts and rules.

Facts are attitudes or properties that are always "true."

Example of properties: ***red*** (apple). Attitude: ***like*** (anne apple).

Rule is a property or Attitude, which is true, if a number of other relations are true.

Rule Format: <fact>: - body <list of truth conditions for the fact>

Example: ***like*** (anne Fruit): - ***red*** (Fruit).

The program execution technique. The program searches for a solution by inference, starting with the first rule, and scanning through the entire list of sentences to determine which sentence should follow the current one. At that, the program selects conditions from the body of the current rule satisfying the conditions of other rules. If all conditions are true, then the rule header is considered to be true and the goal is accomplished. Otherwise, the rule is skipped (goal is not accomplished).

The Visual Prolog interface represents a multi-document window. A text of a separate program is output in a separate window (document). Keywords, text literals, comments, variables, etc. are coloured. Errors found in the course of compilation are output in a separate window. The development language is English. In case the compilation process was successful, the software displays a special window representing a command line.

In Listing 4 we can see the rules from Listing 2 (in Prolog), and in Listing 5 – a log of their execution.

Listing 4. Two rules from ES in Prolog

```
PREDICATES
nondeterm result; nondeterm battery; contacts; ignition_coil
CLAUSES
result        :- write("\nDoes the engine rotate?"),        readchar(Answer),
Answer = 'y'                                                /*if (result)*/
    ,!, contacts.                                           /* then (result)*/
result        :- battery.                                   /* else (result)*/
contacts:- write(" yes\nRun 'Contacts' ").
battery       :- write(" no\nIs the battery charged?"),     readchar(Answer),
Answer = 'y'                                                /*if (battery)*/
    ,!, ignition_coil.                                      /* then (battery)*/
battery :- write(" no\nCHARGE THE BATTERY ").               /* else (battery)*/
ignition_coil         :- write(" yes\nRun 'Ignition_coil' ").
GOAL
result, write("\n\nSolution is found").
```

Listing 5. A Fragment of logging of ES in Prolog

```
Does the engine rotate? n
```

Is the battery charged? n
CHARGE THE BATTERY

Multi Studio

Multi Studio is a hybrid interpreter. Version 082ev Engl (2012).

Basic developer: Center of Intelligent Technologies (Russia), since 2007 [Катаев, 2012].

Global language syntax: nested functions and procedures. Example: ***X (+ (25 3))***.

The logical form and physical structure of knowledge representation (both data and programs) are universal. These are syntactic networks, on which semantic subnets are superimposed. Each network node has a "semant" - a word of Multi, which can be a number, text line, date, name, computer instruction... Instruction is a function (procedure) or its argument. Examples of instructions: ***** (product) ***sin.*** (sine) ***P#*** (arguments alternated with spaces are output onto the computer monitor).

The program is a semantic subnetwork. In general, it includes modules and sentences. A module has a name in the high node and may also consist of modules and sentences. A sentence is a part of the program, finished by '**;**'. The program itself can be a module. The module can consist of a single sentence. Example: ***ZZ (12 34);***

When the source text is input into the system, all the modules are recorded in the knowledge base (container) and are linked in a single network by name. When the program is run it moves along the nodes "from top to bottom and from left to right." This navigation can be dynamically changed by some of instructions ("cycle", "or" ...), as well as by getting a negative result while executing an instruction.

In this language there are several instructions that allow a user to easily develop a query-response (question-answer) system. The principal one among them is ***menu.***; it has the following structure:

menu. ("Header" Option 1 Option 2 ...)

The option structure: field_name (actions)

The field may be a name, text, number, instruction.

The header can be a statement or question of the program to the user. Variants are possible answers (responses) of the user and actions which the program will perform in case this or that option is selected. In case there action list is empty the program goes back to the loop which is closest, and if there are no loops to execute anymore the program ends.

The instruction opens a menu dialog window with the header, fields of options and additional system fields. As an option (answer) is selected the corresponding action is

executed. In particular, that action may be moving to the module which opens a new menu window.

A special case of options is Help. The Help field is a text with an exclamation mark as the very first character. Selecting the Help field opens files specified in the instruction. After the help files are closed, the program automatically returns to the current menu.

Clicking on the additional fields allow the user to return to the previous menu, exit the current loop or quit the program.

The interface of Multi Studio is very diverse, including

- color highlighting of input text,
- syntax error indication (red font color) in the text, up to a single symbol,
- colorful navigation about the program and data during program testing,
- computation result insonification in the human language
- eight system windows with lots of tabs displaying
  - the input text of the program,
  - data under process,
  - the program log,
  - intermediate and final results of its execution
  - help-windows

In addition to the system windows the user program can generate a variety of custom (user) windows (including the form of complex configurations, and menu dialog windows mentioned above).

Multi Studio deals with texts in different natural languages (including Russian). Instructions of Multi are mostly symbolically notated, but the user of Multi Studio can easily change the notation system of by introducing into the dictionary additional synonym instructions.

Example: Listing 6 shows a fragment of the two rules in Multi, which are demonstrated in Listing 2 in terms of CLIPS. Fig. 2 shows the dialog boxes activated by those rules. Listing 8 presents a fragment of the ES log in the Multi Studio environment.

Listing 6. Two rules of ES AUTO in Multi

```
Engine_does_not_start ( menu. ( "Does the engine rotate?"
"yes"(Contacts)                    // if YES then execute "Contacts" module
"no"( Battery);                    // if NO then execute " Battery " module
```

```
Battery ( menu.(  "Is the battery charged?"
"yes"( ignition_coil)          // if YES then execute "ignition_coil "
"no                            // if NO then
CHARGE THE BATTERY";
```

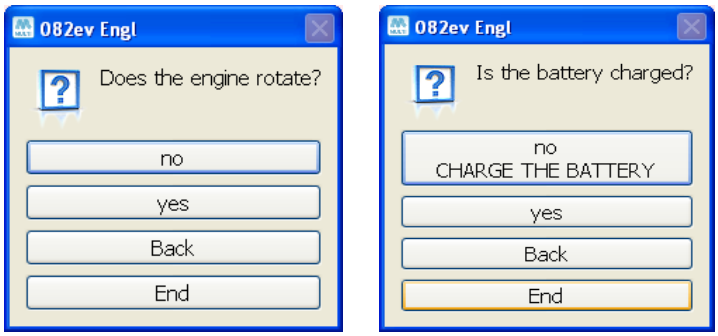


Fig. 2. Dialog boxes of the first and second rules

Listing 8. Fragment of the ES log in Multi Studio environment (two rules)

```
Does the engine rotate?  no
Is the battery charged?  no
CHARGE THE BATTERY
```

Conclusion

The Multi language has the following advantages: universal syntax, higher language level, easiness and conciseness.

As opposed to CLIPS and Prolog, MS is a hybrid interpreter, so MS is faster, more reliable and can process bigger amounts of data and execute big programs. Networks of programs and data get formed as a one-time start compilation is executed, the program forms networks of programs (internal representation), where each rule is linked with the corresponding rules. As a result, when the ES is run the program performs one-time selection of the appropriate rules from the beginning to the decision from the top down, which increases the performance speed (selection speed). CLIPS and Prolog perform a multiple looping searching the rules able to continue the chain. Software performs bottom-up selection of rules and the one-time linking. At that, a conflict may emerge when more than one rule is selected. There are special strategies to be used to solve this trouble.

Besides, Multi allows the user to perform a reverse translation from the internal language to the different national languages (with appropriate dictionaries available which can be created by the user).

The MS interface is more developed compared with the interfaces of CLIPS and Visual Prolog.

Based on the considered ES development environments, we can recommend the development of tool environments in the following directions: development of hybrid systems with a super-high-level universal language and integration of the best qualities of all the languages mentioned above.

Bibliography

[Адаменко и др., 2003] Логическое программирование и Visual Prolog /Адаменко А.Н. [и др.]; - СПб.: БХВ-Петербург, 2003.

[Джарратано, 2007] Джарратано, Джозеф. Экспертные системы: принципы разработки и программирование, 4-е издание. Пер. с англ. / Джарратано, Джозеф, Райли, Гари ; – М. : ООО "И.Д. Вильямс", 2007.

[Катаев, 2012] Катаев, В.А. Разработка экспертных систем в среде Multi Studio / В.А.Катаев // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2012): материалы II Междунар. научн-техн. конф. (Минск, 16–18 февраля 2012 г.); – Минск: БГУИР, 2012, С. 207–212.

Authors' Information

Valentin Kataev *–Intellectual Systems Laboratory Chief, LtD "Perm Scientific Industrial Instrument-Making Company", Perm, Russia;*
e-mail: bravo666666@yandex.ru

Major Fields of Scientific Research: Artificial Intellect, Computational linguistics.