ITHEA

International Journal INFORMATION THEORIES & APPLICATIONS

2005 Volume 12 Number 2

International Journal
INFORMATION THEORIES & APPLICATIONS

ISSN 1310-0513

Volume 12 / 2005, Number 2

IJ ITA is official publisher of the scientific papers of the members of the Association of Developers and Users of Intellectualized Systems (ADUIS).

IJ ITA welcomes scientific papers connected with any information theory or its application.
Original and non-standard ideas will be published with preferences.

IJ ITA rules for preparing the manuscripts are compulsory.
The **rules for the papers** for IJ ITA as well as the **subscription fees** are given on *www.foibg.com/ijita*.
The camera-ready copy of the paper should be received by e-mail: foi@nlcv.net.
Responsibility for papers published in IJ ITA belongs to authors.

General Sponsor of IJ ITA is the **Consortium FOI Bulgaria** (www.foibg.com).

International Journal "INFORMATION THEORIES & APPLICATIONS" Vol.12, Number 2, 2005
Printed in Bulgaria
Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria,
in collaboration with the V.M.Glushkov Institute of Cybernetics of NAS, Ukraine, and
the Institute of Mathematics and Informatics, BAS, Bulgaria.
Publisher: FOI-COMMERCE - Sofia, 1000, P.O.B. 775, Bulgaria. www.foibg.com, e-mail: foi@nlcv.net
® "Information Theories and Applications" is a trademark of Krassimir Markov

ISSN 1310-0513

NEURAL CONTROL OF CHAOS AND APLICATIONS

Cristina Hernández, Juan Castellanos, Rafael Gonzalo, Valentín Palencia

Abstract: Signal processing is an important topic in technological research today. In the areas of nonlinear dynamics search, the endeavor to control or order chaos is an issue that has received increasing attention over the last few years. Increasing interest in neural networks composed of simple processing elements (neurons) has led to widespread use of such networks to control dynamic systems learning. This paper presents backpropagation-based neural network architecture that can be used as a controller to stabilize unsteady periodic orbits. It also presents a neural network-based method for transferring the dynamics among attractors, leading to more efficient system control. The procedure can be applied to every point of the basin, no matter how far away from the attractor they are. Finally, this paper shows how two mixed chaotic signals can be controlled using a backpropagation neural network as a filter to separate and control both signals at the same time. The neural network provides more effective control, overcoming the problems that arise with control feedback methods. Control is more effective because it can be applied to the system at any point, even if it is moving away from the target state, which prevents waiting times. Also control can be applied even if there is little information about the system and remains stable longer even in the presence of random dynamic noise.

Keywords: Neural Network, Backpropagation, Chaotic Dynamic Systems, Control Feedback Methods.

ACM Classification Keywords: F.1.1 Models of Computation: Self-modifying machines (neural networks); F.1.2 Modes of Computation: Alternation and nondeterminism; G.1.7 Ordinary Differential Equations: Chaotic systems; G.3 Probability and Statistics: Stochastic processes

Introduction

In spite of all the achievements of classical physics and mathematics, they have failed to touch upon compete areas of the natural world. Mathematicians had managed to specify, at least, some order in the universe, and the reasons behind this order, but they were still living in an untidy world.

Over the last few decades, physicists, astronomers and economists came up with a way of comprehending the development of complexity in nature. The new science, called chaos theory, provides a method for observing order and rules where once there was only chance, irregularity and, ultimately, chaos. Chaos goes beyond traditional scientific disciplines. Being the science of the global nature of systems, it has brought together thinkers from far apart fields: biology, weather turbulences, the complicated rhythms of the human heart…

Nonlinear and chaotic systems are difficult to control because they are unstable and sensitive to initial conditions. Two close-by trajectories rapidly diverge in phase space and quickly become uncorrelated. Therefore, forcing a system to follow a predetermined orbit is a far from straightforward task. Recently, there have been many attempts at controlling nonlinear and chaotic dynamic systems to get desired phase space trajectories. Ott, Gregogi and Yorke [Ott 1990] proposed one method: a natural unstable periodic system orbit is stabilized by making small time-dependent perturbations of some set of available system parameters. The so-called entrainment and migration control methods proposed by Jackson and Hübler are another approach to controlling chaos [Hübler 1989]. The generalized formulation described by Jackson [Jackson 1990] is based on the existence of some convergent regions in the phase space of a multi-attractor system. In each one of these convergent regions, all the close orbits locally converge to each other. Based on this observation and on many well-researched examples, Jackson stated that every multi-attractor system has at least one convergent region in each basin of attraction. Besides, he described a method for finding such convergent regions. The purpose of the so-called migration goal control is to transfer the dynamics of the system from one convergent region to another. There are many reasons for this. For example, of all the attractors of a complicated system, some can have different types of dynamics (periodic, chaotic, etc.), and one attractor could be more useful for one particular system behavior [Chen 1993].

However, the application of the above control methods has several drawbacks. There must be enough data, control is applied only when the state of the system is very close to the target state, leading to great transitory times closely together before control is activated [Barreto 1995], control is only effective at points adjoining the target state and, after a time, the controlled orbit is destabilized due to the accumulated computational error.

In this article, neural networks are designed to be used as controllers for chaotic dynamic systems, overcoming the problems that appear when using other controller types.

Model of Neural Control

The capacity of neural networks to generalize and adapt efficiently makes them excellent candidates for the control of both linear and non-linear dynamic systems. The objective of a neural network-based controller is to generate a correct control of the signal to direct the dynamics from the initial state to the final target state. The located execution and ease of building a network-based controller depend mainly on the chosen learning algorithm, as well as on the architecture used for control. Backpropagation is used as the learning algorithm in most designs.

The objective of this work is to use neural networks as the structure of a generic model for identifying and controlling chaotic dynamic systems. The procedure that must be executed to control a chaotic dynamic system is shown in Figure 1.

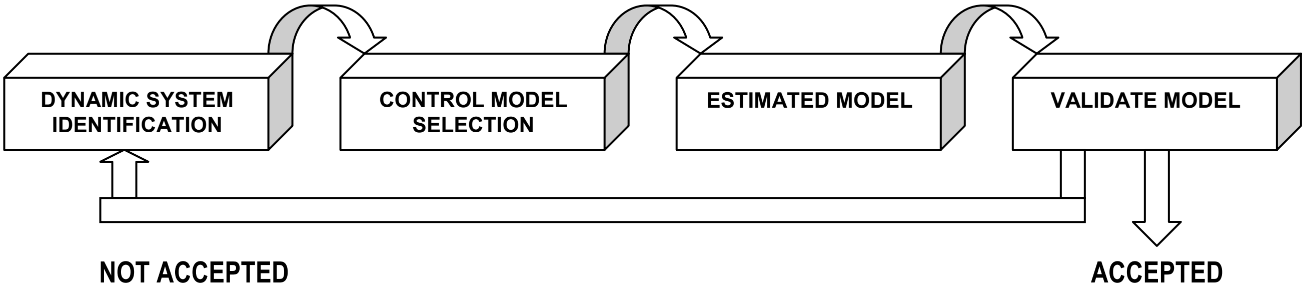


Figure 1: Design procedure of a control model

- Identification of the Chaotic Dynamic System:

This phase involves identifying the system, describing the fundamental data, the operational region, and pattern selection.

$$Zn = \{[\ u(t), y(t)\ ]/t = 1...\ N\}$$

where {u(t)} is the set of inputs, that is, the signal that is to be controlled, {y(t)} represents the output signal, t represents the pattern time. If the system in question has more than one input/output, u (t), and (t) are vectors.

- Selection of the Control Model

Once the data set has been obtained, the next step is to select a structure for the control model. A set of input patterns needs to be chosen, but the architecture of the neural network is also required. After defining the structure, the next step is to decide which and how many input patterns are to be used to train the network.

- Estimated Model

The next stage is to investigate what steps are necessary to make the control effective and to guarantee the convergence of the trajectories towards the target orbit. Control is effective if there is some $\delta>0$ and t_0 such that, for $t > t_0$, the distance between the trajectory and the stabilized periodic orbit is less than δ.

- Validated Model

When training a network, the network has to be evaluated to analyze the final errors. The most common validation method is to investigate residuals (error prediction) by means of crossed validations of a set of tests. The visual inspection of the prediction graph compared with the target output is probably the most important tool.

Identification of the Chaotic Dynamic System

The systems that are going to be controlled are nonlinear and chaotic dynamic systems that depend on a system of parameters, p. The basic function is: $\frac{dx(t)}{dt} = F(x(t), p)$, where F: $\Re^n \to \Re^n$ is a continuous function.

The other type of systems investigated is discrete dynamic systems, represented by an equation of nonlinear differences. Such systems are described as a function f: X → X that determines the behavior or evolution of the set when time moves forward. The control system inputs are the orbits of the elements. The orbit of $x \in X$ is defined as the succession $x_0, x_1, x_2...., x_n...$, achieved by means of the rule: $x_{n+1} = f(x_n)$ with $x_0 = x$

The points of the orbit obtained are:

$x_1 = f(x_0) = f(x)$; $x_2 = f(x_1) = f(f(x)) = f^2(x)$; $x_3 = f(x_2) = f(f^2(x)) = f^3(x)$;… $x_n = f(f(... f(x)...)) = f^n(x)$ n times

The behavior of the orbits can vary widely, depending on the dynamics of the system.

The objective is to control the dynamic system in some unstable periodic orbit or limit cycle that is within the chaotic attractor. Therefore, the output of the system will be the limit cycle of period-1 or greater in which the system must be controlled. To find the outputs, it is necessary to consider that:

- A point α is an attractor for the function f(x) if there is a neighborhood around α such that the point orbits in the neighborhood converge to a. In other words, if the values are near to α, the orbits will converge to α.
- The simplest attractor is the fixed point. A point α is a fixed point for the function f(x) if f(α) = α.
- A point α is periodic if a positive integer number τ exists such that $f^\tau(\alpha) = \alpha$ and $f^t(\alpha) \neq \alpha$ for 0<t<n. The integer τ is known as the period of α.
- If α is a fixed point attractor, the set of initial values x_0 whose orbits converge to α form the basin of attraction of α.

The discrete systems that have been controlled are:

Discrete Systems:

Systems $f : R^2 \to R^2$ are second-order controlled, nonlinear discrete systems. All the trajectories are directed towards the stable point $x_{n+1} = f(x_n)$, where $x_n \in R^2$. The chosen systems are the Henon [Martin 1995], Lozi [Chen 1992], Ikeda [Casdagli 1989], and Tinkerbell [Nusse 1997] systems. The same type of discrete systems are controlled in unstable periodic orbits (limit cycle). The systems used in this case are Ikeda and Tinkerbell.

Another approach to controlling chaos is the so-called entrainment and migration control methods proposed by Jackson [Jackson 1991] and Hübler. Gumowski and Mira's discrete system is used to achieve migration control. This system has several attractors and several bounded convergent regions in the basin of attraction.

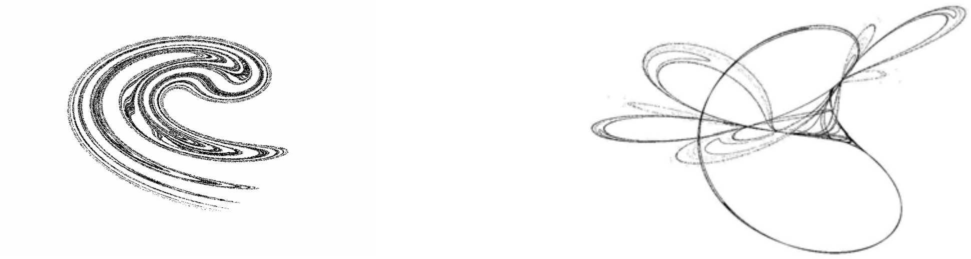


Figure 2. Ikeda system trajectory and Tinkerbell map

Continuous Systems: Continuous systems $f : R^3 \to R^3$ are controlled in an orbit of period 1, that is, at an equilibrium point. We look at Lorenz's [Gulick 1992] and Rössler's systems.

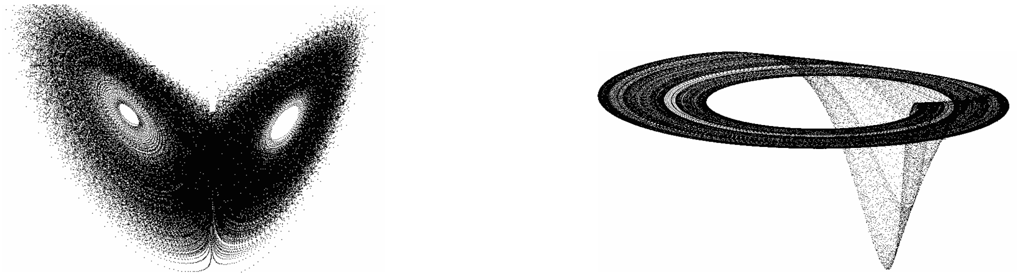


Figure 3. Lorenz's Attractor Rössler's Attractor

Selection of the Control Model

The structure of the model has been divided to address two sub-problems:

- design the network architecture
- choose the structure of the input patterns

Architecture of the Neural Network

According to Lippmann [Lippman 1987], a model of neural network is characterized by specifying:

- The transference function of each node
- The network topology, which is defined by the number of nodes and the set of interconnections between these nodes.
- The learning rules, which are the rules that regulate how the weights associated with the connections are found.

The transference function. The neuron activation function is the sigmoid.

The neural network topology. The neural network employed as the main controller is composed of three layers of neurons (input layer, hidden layer and output layer).

Input layer:

- When control is effected at an equilibrium point, the input layer has two neurons, one for each of the variables of the function f that is going to be controlled for the discrete functions $f : R^2 \to R^2$, plus three neurons, one for each of the variables of the function f that is going to be controlled for the continuous functions $f : R^3 \to R^3$.
- When control is effected in a limit cycle, the components of the vector are separated from the function f, which will be applied first to the first and then to the second component of the function. The input layer will have as many neurons as the period of the limit cycle. If control is effected in a limit cycle of period 7, the input layer has 7 neurons; if the period is 5, the input layer has 5 neurons.

Output layer:

- When control is effected at an equilibrium point, the output layer has two or three neurons corresponding to the coordinates of the stable point.
- When control is effected in a limit cycle, the output layer will have as many neurons as the period of the limit cycle. If the period of the output layer is 7, it will have 7 neurons that correspond to the first coordinates of the period-7 point.

The number of neurons in the hidden layer plays an important role in the learning performance and generalization capability of the network:

- For the discrete functions $f : R^2 \to R^2$, the hidden layer will have one hidden neuron when control is effected at an equilibrium point.
- For the continuous functions $f : R^3 \to R^3$, several simulations have been run in order to ascertain how the number of hidden neurons affects the mean square error in finding the stable point.

The learning rules. The algorithm that is going to be used to adjust the weights is backpropagation, which descends according to the gradient that minimizes the error function.

Structure of the input patterns

Input patterns are necessary to define appropriately it, to avoid one of the biggest problems: trapping in a local minimum. The patterns for training the network will be formed by system orbits, obtained starting from a point that is within the basin of attraction of the limit cycle chosen for controlling the system.

1. Input Patterns

The input patterns are obtained by taking a starting point (x_0, y_0) and finding the time series from the components of the function by iterating $x_{n+1} = f(x_n)$. The input file is constructed from one point, using 500 patterns to search the time series.

When control is effected in a limit cycle, for example, a period-5 limit cycle, input patterns will be constructed as follows. The first pattern will be constructed from a point A= (x_0, y_0): {x_0, f(x_0), $f^2(x_0)$, $f^3(x_0)$, $f^4(x_0)$ }, the following one from the next five iterations, and so on.

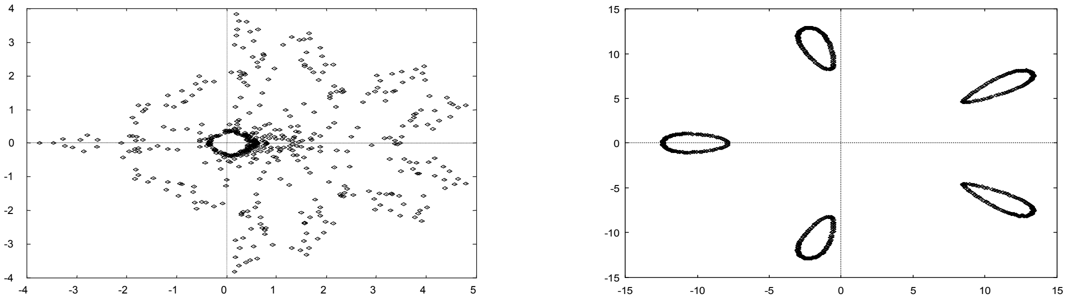


Fig. 5. a) Input for the network init point (0.5, 0) b) Input for the network init point (-2.37, 12.83)

Figure 5 shows the network input, that is, the input file consisting of 500 points from the iterative Gumowski and Mira function starting from points of different basins of attraction. The number of iterations is 10, and the final mean squared error after the learning process is 0.009397477, a good threshold.

2. Output Patterns

The input pattern is the equilibrium point at which the control function is going to be controlled.

When control is effected in a limit cycle, for example, a limit cycle of period 5, the output layer will have five neurons that correspond to the first coordinates of the period-5 point. If the point is Q = (q_1, q_2), the output will be constructed as {q_1, f(q_1), $f^2(q_1)$, $f^3(q_1)$, $f^4(q_1)$}, which is the orbit of period 5.

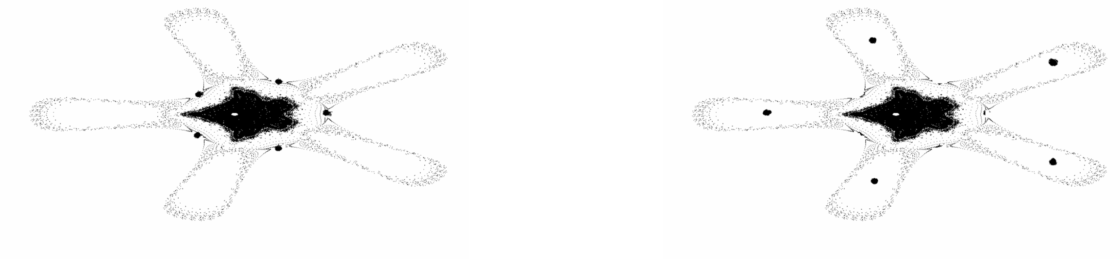
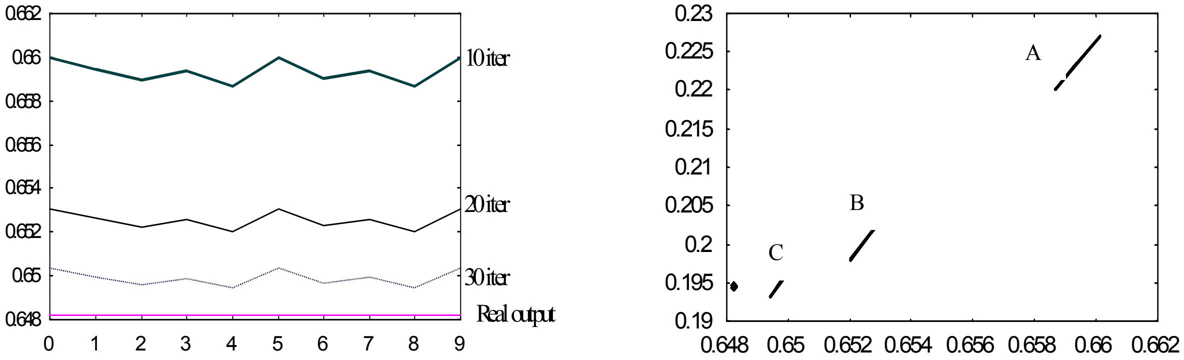


Fig. 6. 5-period orbits of the Gumowski and Mira function located in different basins of attraction

Estimated Model

Once the control model has been designed, the following steps are taken:

1. Network weights must be fixed and always have same values to assure deterministic behavior.
2. An input pattern is presented and the output is calculated. To finish the learning phase of the network, another input pattern set is output starting from a point and finding the time series with 500 function patterns. The number of iterations to be learned is 10, and the mean square error is acceptable. Therefore, the network has found a good solution.
3. Number of input patterns. The error variance has been studied across the number of input patterns, including files with different patterns. However, the error decreases considerably as the number of sweeps increases. This is due to the fact that the network is forced to learn the patterns across more iterations. Accordingly, the error is approximately the same if the total number of patterns used to train the network ties in. Therefore, if there are few data, the number of network iterations in the learning phase needs to be increased for the error to be considered acceptable.



a) b)

Figure 7. Real outputs from the Henon system network

Figure 7 a) shows that when the number of iterations increases, then the output of the network will better approximate the target output, that is, the stable point 0.648. One of the more important advantages of this technique is that the controllers obtained are very stable even with low random dynamic noise or with few data. Figure 7 b) shows the behavior of the Henon map with randomly added dynamic noise, controlled at the stable point and with 30 iterations in the learning phase.

Validate Model

Once the learning phase is complete, it is necessary to check if the network is able to control the function at the stable point.

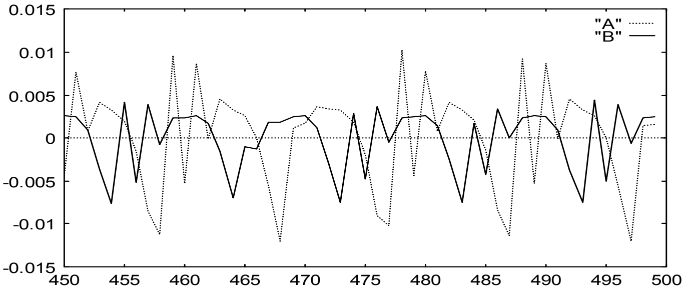


Figure.8. Real outputs of the network of the first component of the controlled Tinkerbell function around each point of the period-5 orbit (initial points A and B)

The error will be checked for acceptability, because the network needs to be validated with different orbits and the outputs and the errors need to be examined. It is also necessary to verify that the control model is robust, that is, to study the network errors when there is any sort of noise in the patterns. The fastest way of verifying the network output errors is by means of a graph.

Applications

The control models have been designed according to the previously described neural networks-based procedure to control several systems in unstable periodic orbits. Second-order discrete non-linear systems have been controlled at an equilibrium point. The chosen systems are the Henon and the Lozi systems [Hernandez 1999]. Another discrete system, Ikeda, was controlled [Hernandez 1999] at equilibrium point and a period-5 point, that is, in a period-1 and a period-5 limit cycle. The Tinkerbell system [Hernandez 2000] was controlled in period-1, 5, and 7 limit cycles. Later, the continuous Lorenz system [Hernandez 2001] was controlled by means of the same procedure.

Also, the same control model has been applied to a dynamic system with multiple attractors, and a controller has been designed to transfer and control the orbits of any basin of attraction of the different system attractors at an equilibrium point and in a period-5 limit cycle. The chosen system is Gumowski and Mira's system [Hernandez1 2001]. Finally, two chaotic signals for both discrete and continuous systems were simultaneously controlled [Hernandez 2002], [Hernandez1 2002].

Conclusions

The main contributions of this work are:

- The construction of a controller model that uses a neural network which is very simple to design.
- The control method is flexible, it can be adapted to any system and can even to control and classify several systems simultaneously.
- The control can be applied to any point of the system, there being no need to wait for the system to approach the control target orbit.
- Control is effective for as long as it is applied, the accumulation of computational errors has no influence, whereas, in other methods, it destabilizes the system after a number of iterations.
- Control is robust, its behavior is satisfactory even in the presence of random dynamic noise and even if there are few data, which is very important due to chaotic systems' sensitivity to initial conditions.
- The systems use the same model both to control a system with a single attractor in an unstable periodic orbit and to control a multi-attractor system in an unstable periodic orbit of any basin of attraction, transferring the system dynamics from one attractor to another.
- The learning speed of the designed control model is high, and few errors are generated.

The networks used for the control model have supervised learning. Other types of networks, such as associative networks, might be investigated. It is also worth looking at the possibility of controlling chaotic systems in which attractors are not known by training the network with orbits of the different basins of attraction.

References

[Barreto 1995] 'Multiparameter Control of Chaos' Ernest Barreto and Celso Grebogi, Physical Review E. Volº2 nº4, pp 3553-3557, (1995)

[Casdagli 1989] "Nonlinear prediction of Chaotic time series" , Casdagli M., Physica D35, 335-356, 1989.

[Chen 1992] "On Feedback Control of Chaotic Dynamical Systems", G. Chen y X. Dong, Int. J. of Bifurcations and Chaos, 2, pp 407-411, (1992)

[Chen 1993] "From Chaos to Order", Chen G. and Dong X., *Int. J. of Bifurcations and Chaos 3*, Pp. 1363-1409, 1993.

[Gulick 1992] 'Encounters with Chaos', D. Gulick, McGraw-Hill, Inc 1st edition, (1992) [Hübler 1989] 'Adaptive control of chaotic systems', A.W. Hübler, Helvetica Physica A62, pp 343-346, (1989)

[Jackson 1990] '*The entrainment and migration controls of multiple-attractors systems'*. Jackson, E. A., Phys. Lett, A, 151, Pp. 478-484,1990.

[Jackson 1991] 'Entrainment and migration controls of two-dimensions maps', E. A. Jackson and A. Kodogeorgion, Physica D54, pp 253-265, (1991)

[Hernandez 1999] "Neural Network Control of Chaotic Systems*". Hernandez, C., Martinez, A., Castellanos, J., Computational Intelligence for Modelling, Control &Automation. Concurrent Systems Engineering Series. ISSN: 1383-7575. Vol. 54, pp. 1-8. 1999.*

[Hernandez1 1999] "Controlling Chaotic Nonlinear Dynamical Systems". *Hernández C., Martínez A., Castellanos J., Mingo L.F..: IEEE Catalogue Number 99EX357. ISBN: 0-7802-56X2-9. Piscataway N.J. USA. pp. 1231-1234. 1999.*

[Hernandez 2000] "Periodic Orbit Stabilization with Neural Networks". *Hernández C., Martínez A., Mingo L.F., Castellanos J.; Frontiers in Artificial Intelligence and Applications. Vol. 57. New Frontiers in Computational Intelligence and its Applications. IOS Press. ISSN: 0922-6389. pp. 109-118. 2000.*

[Hernandez 2001] "Migration Goal Control of Chaotic Systems with Neural Networks". *Hernandez, C., Castellanos, J., Martinez, A., Mingo L.F.; Knowledge Based Intelligent Information Engineering Systems and Allied Technologies. Frontiers in Artificial Intelligence and Applications. IOS Press Ohmsha. ISSN: 0922-6389. ISBN: 1-58603-1929. Vol.: 69. Part II. pp.: 1165-1169. 2001.*

[Hernandez1 2001] "Controlling Lorenz Chaos with Neural Networks". *Hernandez, C., Martinez, A., Mingo, L.F., Castellanos, J.; Advances in Scientific Computing, Computational Intelligence and Applications. WSES Press. ISBN: 96-8052-36-X. pp. 302-309. 2001.*

[Hernandez 2002] "Neural Control of Simultaneous Chaotic Systems". *Hernandez, C., Gonzalo, R., Castellanos, J., Martinez, A.; Frontiers in Artificial Intelligence and Applications. Vol. 82. IOSPRESS. ISSN: 0922-6389. pp. 527-531. 2002.*

[Hernandez1 2002] "Simultaneous Control of Chaotic Systems". *Hernández, C., Martinez, A., Castellanos, J., Luengo, C.; Recent Advances in Circuits, Systems and Signal Processing, WSEAS Press. ISBN: 960-8052-64-5. pp. 200-204. 2002.*

[Hübler 1989] '*Adaptive control of chaotic systems',* Hübler, A.W., Helvetica Physica A62, Pp. 343-346, 1989.

[Martín 1995] 'Iniciación al caos', Miguel Ángel Martín, Manuel Morán, Miguel Reyes; Editorial Síntesis, ISB: 84-7738-293-X, (1995)

[Nusse 1997] "Dynamics: Numerical Explorations", H.E. Nusse, J.A. Yorke, J.E. Marden and L. Sirovich (Springer-Verlag, New York). Series: Applied Mathematical Sciences V. 101, (1997)

[Lippmann 1987] 'An Introduction to Computing with Neural Nets' L. P. Lippmann, IEEE ASSP Magazine, April 1987 pp. 4-22, (1987)

[Ott 1990] '*Controling Chaos* ', Ott E., Grebogi C. & Yorke J. A.,. Phys. Rev. Lett, 64, Pp. 1196-1199, 1990.

Authors' Information

Juan Castellanos Peñuela- Departamento de Inteligencia Artificial, Facultad de Informática – Universidad Politécnica de Madrid (Campus de Montegancedo) – 28660 Boadilla de Monte – Madrid – Spain; e-mail: jcastellanos@fi.upm.es

Cristina Hernández de la Sota- Departamento de Inteligencia Artificial, Facultad de Informática – Universidad Politécnica de Madrid (Campus de Montegancedo) – 28660 Boadilla de Monte – Madrid – Spain; e-mail: cristinah@renfe.es

Rafael Gonzalo Molina- Departamento de Inteligencia Artificial, Facultad de Informática – Universidad Politécnica de Madrid (Campus de Montegancedo) – 28660 Boadilla de Monte – Madrid – Spain; e-mail: rgonzalo@fi.upm.es

Valentín Palencia Alejandro- Departamento de Arquitectura y Tecnología de Sistemas Informáticos, Facultad de Informática – Universidad Politécnica de Madrid (Campus de Montegancedo) – 28660 Boadilla de Monte – Madrid – Spain; e-mail: vpalencia@fi.upm.es

BRIDGING THE GAP BETWEEN HUMAN LANGUAGE AND COMPUTER-ORIENTED REPRESENTATIONS

Jesús Cardeñosa, Carolina Gallardo, Eugenio Santos

Abstract: *Information can be expressed in many ways according to the different capacities of humans to perceive it. Current systems deals with multimedia, multiformat and multiplatform systems but another « multi » is still pending to guarantee global access to information, that is, multilinguality. Different languages imply different replications of the systems according to the language in question. No solutions appear to represent the bridge between the human representation (natural language) and a system-oriented representation. The United Nations University defined in 1997 a language to be the support of effective multilinguism in Internet. In this paper, we describe this language and its possible applications beyond multilingual services as the possible future standard for different language independent applications*

Keywords: *Knowledge Representation, Information modelling*

ACM Classification Keywords: *H.3.3 Information Search and Retrieval; I.2.7 Natural Language Processing:* Machine translation; *I.2.4 Knowledge Representation Formalisms and Methods:* Representation languages; *K.4.4 Electronic Commerce:* Distributed commercial transactions

Introduction

The concept of Multimedia systems could be broadly defined as the set of systems built for transmitting information through any means that human beings are able to catch. People communicate in a natural way through the five senses, either individually or cooperatively. The evolution of technology has permitted the existence of systems in the market that reproduce the natural five senses. Among these systems and technologies, the most classical and frequent systems to transmit information utilize sight and hearing. These natural senses have permitted to see and hear. They have allowed us to visualize images of all kinds, from natural images to drawings and pictures, and from static to animated images. Further, we have been able to perceive them combined with natural or artificial sounds. However, sight and hearing do not constitute the most massive means to transmit information; the most appealed media to transmit knowledge among human beings is language, mainly recorded in its written form.

Ever since immemorial times, when human beings wanted knowledge to endure, they recorded in written texts. Even today, in the image era, images are accompanied by text. The profusion of support media for information has made that classical systems based on natural language texts incapable of organizing the information in an adequate manner, hindering a correct management of information. Several technologies like document management or information retrieval have been helped by great computational systems in order to palliate such situations.

From the second half of the XX century, when the production of written information has been massive, Internet has put in an appearance, and worldwide commerce has grown exponentially. Media companies have started to commercialize information by itself, which means that they have to conceive systems for storing information in an organized and more compact way, easier to find and thus easier to be offered to those requesting it. This requires of complex systems but also of ways of representing knowledge that permits the reliable interchange of information between humans and machines. Human language is the externalization of human knowledge and the vehicle for knowledge exchange among humans but it is inadequate for machines.

An intelligent use of knowledge for any purpose (like question answering, information retrieval, concepts deduction, etc.) calls for mechanisms of representation completely devoid of any source of ambiguity and imprecision found in natural languages, while endowed with the formal properties characteristic of computational languages.

For that, we should invest more resources in the processes of capturing, organizing and diffusing information. Thus, in the same way that we associate *knowledge* to the "sources", we should ascribe *information* to "systems", i.e., to the media in charge of providing information. That is, *knowledge* and *information* is not the same thing, and the transformation of one into the other entails something more that human language. It entails a system that permits the conversion of one into the other.

However, the design of an intermediate system between human language, as a way to represent knowledge, and systems language is not a trivial issue. In fact, human language is self-organized in an unconscious manner; it represents highly complex mental processes (be it in written or oral form) such as searching and rearranging data in a way that can are useful and comprehensible for others. On the other hand, systems for the capture and diffusion of information have a conscious organization but an information search capability quite limited. That is:

	Knowledge Organization	**Functionalities**
Human language	Unconscious	High
Systems language	Conscious	Low

The gap between the organization of human knowledge and systems knowledge still remains as the main obstacle for the construction of efficient systems with conscious knowledge, able to be easily maintained, modified or expanded. A possible approach to "fill this gap" could be an intermediate representation of knowledge (hereafter *IR*) serving as a bridge between human language and systems language. That is, a representation able to express contents with schemata and models close to human beings but at the same time a representation able to communicate with systems or represent knowledge at the systems level. Figure 1 illustrates the intermediary role that the IR language could play among human languages and systems language.

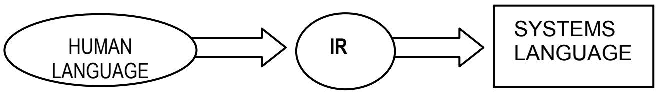


Figure 1. Intermediate Language between human language and systems language

There have already been attempts to develop these languages, although not associated to the use that we propose here. We are referring to Schank's theory of Conceptual Dependencies [Schank, 1972] to other models such as Sowa's [Sowa, 1996] that have been in fact predecessors of current ontologies. However, these models lacked a number of applications and a wider computational capability when they were proposed. In another sector of the industry, concretely language industry, we can find similar models used to represent contents in a language-independently manner. These models were known as "Interlinguas" and were used in machine translation systems such as PIVOT [Muraki, 1989] and ATLAS [Uchida, 1989].

We could say that, although with a different perspective, there are already precedents of languages and models that have attempted to overcome the gap between human languages and machine languages. We will generically call *Intermediate Representation Language* to the knowledge representation language able to make compatible human languages and systems languages.

In the late 90ies, the University of the United Nations started the development of a computational language that would allow for the representation of human knowledge in a language-independent way. The rationale behind this was to the elimination of linguistic barriers for content access in Internet. The mere fact of guaranteeing language independence involved the representation of the deepest structures of conscious human expression, so that written texts in any natural language would have the same representation in that computational language. After years of research and testing, such language has been reaffirmed as a language able to be understood by systems (for example, in order to generate any other human natural language) and it is possible to generate it just following its specifications and corresponding manuals [Uchida, 2003]).

This approach is resulting to be much more important that the initially posed (pivotal representation of texts for the elimination of linguistic barriers). This language has been gradually transformed into a knowledge representation language starting from written texts.

In this article it is presented a description of this language and how from written texts we can achieve a representation that not only "stores" knowledge but allows for an intelligent use of the resulting representation for different purposes, provided a number of functions.

The Language

UNL is an artificial language designed for unambiguously expressing the informational content of natural language texts in a language-independent way, with the main aim of facilitating automatic multilingual information exchange on the web or in other local contexts.

Information encoded in UNL is organised into UNL documents. Since documents are commonly organised themselves into paragraphs and these into sentences, a UNL document mimics such structure and is organised into UNL paragraphs and sentences by means of HTML-like tags. UNL paragraphs consist of UNL sentences, which in turn are divided into sections. Each UNL sentence is divided into the following sections:

- The original sentence, i.e. the information that has been encoded.
- The UNL code corresponding to the original sentence.
- For each language for which a UNL Generator has been developed the automatically generated text of the UNL code into that language. Generation results are then cached in the document and available to the reader without delay. Of course, the stored results can be renewed as soon as the generators improve their output.

Besides these elements, a UNL header contains information and meta-information about the document as a whole. Although not devised in principle, UNL documents also allow for its integration into XML, since document structuring such as tables, sections, subsections, etc are not easily expressed by the UNL machinery. In fact, it is not the objective of UNL to provide document structuring but to provide a semantic structuring of contents. These ideas are further explored in [Cardeñosa, 2005] and [Hailaoui, 2005].

A UNL expression takes the form of a directed hyper-graph. Its simple nodes contain the so-called *Universal Words* and its arcs are labelled with *conceptual relations*. In addition to simple nodes, hyper-nodes are also allowed as origin or destination of arcs and consist on UNL graphs themselves. In addition to universal words and conceptual relations, *attributes* are the third ingredient of a UNL hyper-graph. Attributes may occur as labels of the universal words, modifying them in certain key aspects. These are the three building blocks of the inter-lingua and we now turn to describe each one in detail.

Universal Words

Universal words are so called because they attempt to be universally applicable to any natural language and because their meaning is derived from the meaning of natural language words. UWs are based on the English headwords. Initially any English headword is a candidate for becoming an UW, being its meaning the meaning defined in any authoritative monolingual dictionary of English. However, in deciding on English headwords as the building blocks of the vocabulary of the interlingua, two key aspects have to be resolved:

a) Inherent ambiguity in English headwords.
b) Mismatch among lexicalized concepts in English and lexicalized concepts in other natural languages.

Inherent Ambiguity in English Headwords

Most natural language words are subject to ambiguity and polysemy. A single word in a natural language contains several senses (often related, often not), so it is fairly rare that the relation between a concept and a word is one-to-one. English vocabulary is not devoid of such ambiguity and thus a system based on English headwords as interlingual concepts should establish mechanisms for reducing such ambiguity. In order to reduce ambiguity, UWs are modified by semantic restrictions. Such semantic restrictions try to select a given sense or concept of an English Headword from the others. The most basic and simple way to achieve this is by attaching to the word a hypernym.

We will illustrate the process of defining UWs taking the following sentence as input:

```
Member States should, whenever necessary or desirable, conclude
bilateral agreements to deal with matters of common interest arising out
of the application of the present Recommendation.
```

Only content words (mainly nouns, verbs, adjectives and some prepositions and adverbs) require an UW, in this sentence, the first content word is *State*, which is a highly ambiguous headword in English.

State can be both a noun and a verb in English. Initially, there are there are two obvious candidates for "state" (being "icl" the abbreviation for "is included in" or the traditional *"is a"* relation) using the most general hypernym as semantic restriction:

statet(icl>thing) → for the nominal senses
state(icl>do) → for the verbal senses

However these restrictions are not very informative and there is still a great degree of ambiguity in each of these potential UWs. In order to overcome this ambiguity, more and finer semantic restrictions have to be attached to the basic UWs. For this, the possibilities are the following:

1. Use of a closer hypernym.
   a. state(icl>government)
   b. state(icl>region)
   c. state(icl>circumstance {>abstract thing})
2. Use of argument structure (for verbal senses):
   a. state({icl>do} agt>thing, obj>thing)

Lexical Mismatches among Languages

This is the second problem that in fact, *every IL has to tackle*, and in the context of UNL, the solution does not come at first sight. Lexical mismatches arise when:

a) For a given concept in a given language, there is not English headword. This is a frequent case for cultural-specific terms (and other not so cultural-specific).
b) For a given concept in English, there is not an appropriate term in the target natural language. Example: En. misunderstand → Sp. entender mal
c) There is not a one-to-one relation among languages, an English headword is covered in the target language by more than one headword:
   a. Corner → Sp. esquina & rincón
   b. Marry → Ru. zhenit'sja & vyxodit' zamuzh

For these three situations, a solution must be provided. So, the solution adopted in each case is the following:

a) For the lexical gap in the English language, the specifications of the interlingua propose to include the original word as the basic UW and then semantic restrictions would be added to describe the intended meaning as much as possible, like in the Japanese term *Ikebana(icl>flower arrangement)*.
b) Lexical gaps in the target languages can be considered as a "local" problem, to be treated in the dictionaries of target languages, and not in the design of the Interlingua. For example, in the case of "misunderstand" and Spanish, it will be the task of Spanish developers' dictionary to link such a word with a complex expression such as "*entender mal*".
c) When an English headword combines the meaning of more than one headword in another language, it is possible to appeal to the semantic restrictions again in order to clarify the intended concept. In fact there are two possibilities:
   a. Ignore the difference, the Anglo-centered vocabulary here is imposed, and it will be the task of local generator to choose (in the case of Russian) into the verb *vyxodit' zamuzh* or the verb *zhenit'sja*.
   b. Express the difference, with the use of the semantic restrictions like for example:
      i. *[vyxodit' zamuzh] marry(agt>female); [zhenit'sja] marry(agt>male)*
      ii. *[esquina] corner(mod>outside angle); [rincón] corner(mod>inside)*

Of course it will be desirable that all these UWs conforms a hierarchy of inter-related UWs, so that *marry(agt>female)* and *marry(agt>male)* depend of a more general UW like *marry(agt>person, obj>person)*. Thus, semantic restrictions impose themselves a hierarchy into the system of UWs. The result is the so-called UNL-KB organizing the UW concepts *à la Wordnet* [Fellbaum, 1998], thus implicitly linking and relating the vocabulary of natural languages through the pivotal UW system.

UNL Relations

The second ingredient of UNL is a set of conceptual relations. Relations form a closed set defined in the specifications of the inter-lingua. The rationale behind conceptual relations is twofold:

1. To characterize a set of semantic notions applicable to most of the existing natural languages. For instance, the notion of initiator or cause of an event (agent) is considered one of such notions since it is found in most languages.
2. To select a small set of semantic notions relevant to produce an inter-lingual semantic analysis. The notion of agent is regarded as one of such relations, because of its central role in the analysis of the meaning of many sentences.

Therefore, a UNL representation is mainly based on a role-based description of an event or situation, following the tradition started by Fillmore's case grammars, rather than a more logical semantic analysis of sentences. When defining the intended meaning of each conceptual relation, the specifications of the language [Uchida, 2003] rely on two intensional expedients:

1. Setting semantic constraints over the universal words that can appear as first (origin) and second argument (destination) of the relation. These constraints are based on the lexical relations established among universal words in the Knowledge Base. In the case of the agent relation, such restrictions include that the origin Universal Word must denote an event that accepts an initiator (and not an event that "just happens") and the destination universal word must denote an entity (as opposed to a quality or an event, let us say).
2. Giving a natural language explanation of the intended meaning of the relation. For the agent relation this explanation may just say that the agent is the cause or initiator of the event, an entity without which the event would not happen.

Provided that the Knowledge Base is built upon unambiguously defined principles, the first mechanism gives us a rigorous characterization of the conceptual relations. The second expedient is subject to the ambiguities of natural language, and the resulting definition is therefore semi-formal.

The current specification of UNL includes 41 conceptual relations, including causal, temporal, logical, numeric, circumstantial and argument relations.

Selecting the appropriate conceptual relation plus adequate universal words allows UNL to express the propositional content of any sentence.

In the input sentence, some of these relations are exemplified. This sentence describes a main event, denoted by the English predicate *conclude* and its dependent participants. The UW for the main predicate of the sentence is *conclude(agt>thing, obj>agreement)*. It requires two participants:

a) The agent or initiator of the event: In this sentence, the agent is "Member States" that coincides with the subject of the clause.
b) An object (or theme affected by the event, required for the completion of the event), realized in the "bilateral agreement" as the direct object.

Given the syntax of UNL and the binary nature of relations, when specifying the UNL representation of the sole event "Member States should conclude bilateral agreements" the *agt* relation links *conclude(agt>thing, obj>agreement)* as source UW and *state(icl>government)* as target UW. Analogously, there is a modifying *mod* relation between *state* and *member*; and *agreement* and *bilateral.*

Figure 2 shows a graphical version of the UNL representation of the main clause of the sentence.

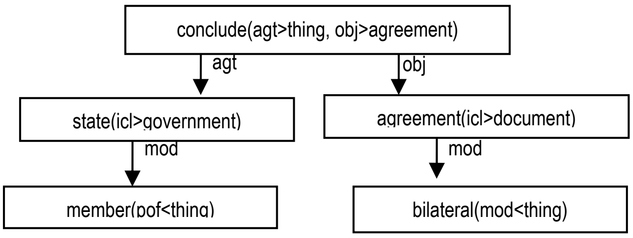


Fig. 2. Representation of the main event of the sentence

Description of Attributes

Contextual information such as the time of the event with respect to the time of the utterance, informative structure of the sentence, speaker's communicative goal and attitudes, etc. is expressed in UNL by means of *attribute labels*. UNL attributes include notions such as:

- Information depending on the speaker, such as time of the described event with respect to the moment of the utterance; the communicative goal of the utterance; epistemic and deontic modality.
- Contextual information affecting both the participants both the predicate of the sentence, like aspectual properties of the event, number of nominal concepts.
- Pragmatic notions like the organization of the information in the original sentence, referential status of referring expressions and other labels determining discourse structure.
- Typographical and orthographical conventions. These include formatting attributes such as *double quotations, parenthesis, square brackets*, etc.

Applications

The UNL System has an indubitable application to all the existing information systems. The introduction of multilinguality in any other system is almost a model case of added value services. However, it does not mean that multilinguality be only translation services. We will describe some possible applications of the UNL system.

UNL as Language for Knowledge Representation

UNL is mainly used as a support language for multilingual generation of contents coming from different languages. However, its design allows for non-language centered applications, that is, UNL could serve as a support for knowledge representation in generic domains. When there is a need to construct domain-independent ontologies, researches turn back to natural language (such as Wordnet, GUM [Bateman, 1995] or even CyC [www.cyc.com]) to explore the "semantic atoms" that knowledge expressed in natural languages is composed of. UNL follows this philosophy, since it provides an interlingual analysis of natural language semantics. The reasons why UNL could be backed as a firm knowledge representation language are:

1. The set of necessary relations existing between concepts is already standardized and well defined.
2. It is the product of intensive research on the thematic roles existing in natural languages by a number of experts in the area of Machine Translation and Artificial Intelligence, guaranteeing wide coverage of all contents expressed in any natural language.
3. Similarly, the set of necessary attributes that modify concepts and relations is fixed and well defined, guaranteeing a precise definition of contextual information.
4. UNL syntax and semantics are formally defined.

However, to really serve as a language for knowledge representation, it must support deduction mechanisms and must specify how a knowledge base could be build up in the UNL language. This idea is explored in [Cardeñosa, 2004].

Being a language suitable for knowledge representation, UNL could be the support of ontologies or of Cross Language Information Retrieval systems. UNL could be a firm candidate for this because of its long history as an interlingua, and the existence of analysis and generation systems to and from UNL.

Cross-lingual Information Retrieval

To support cross-lingual information search could be one of the most appealing applications. Because the information existing in UNL is in fact independent of the original language, placing information in a web site written in UNL supposes that is accessible from any other language. But also, the search systems could try to find information based on concepts (much more effective than based on terms or keywords that are dependent on the language) and find it (if it exists) independently of the language used by the generator of the information searched. UNL offers a promising approach to this kind of systems because the search of information based on concepts is not difficult to be re-written in UNL (they could be UW) almost automatically, always under the supervision of the searcher.

Multilingual Information System

One possible situation is that an organization has public information that should be shared and distributed in multilingual form. This is not exactly a problem of translators (at an acceptable cost) but a problem of maintenance. In this case, an organism, such as any derived from the United Nations for instance (or any other as the European Commission, Health Care Organisation, etc.) should maintain an on-line system with all kind of information about organizations. This could be classified as a simple multilingual service, where the information should be written in UNL (in the UNL Document Base) and shown through the Web in different languages. The maintenance is carried out by making changes in the UNL code. The style in which organisations are described makes post-editions unnecessary most of the times.

An additional use of this kind of system is the maintenance of technical documentary databases with multilingual necessities. One case would be the technical documentation (maintenance of industrial equipment for instance) that has to be managed in many countries, or in the case of companies with branches in several countries where a clear and precise documentation is essential for reaching a leading market position. Writing this technical documentation in UNL would clearly permit unified contents so that no differences derived from different translators could cause technical problems. It is well known that the manuals for some languages are almost in all cases translated from the original language into English and from English to the target language introducing in some cases a double risk of mistakes. The use of the UNL system for this kind of application permits also that the post-edition (if needed) can be done directly by the final users.

The multilingual Access to Public Sector Information is a general goal of big public organizations. One of the major problems to reach the objective to make the public information really available is the multilingual origin of the documents. In fact one of the recommended actions mainly to the European Industry is to affiliate contents to the Multilingual e-Content Europe portal [Nicholas, 2000]. Of course, having multilingual contents, the industry and public organizations have also to guarantee the multilingual access.

Multilingual Transactional System

Different issues have been solved in the last years to facilitate the Business to Customer (B2C) services as the typical practice of Electronic Commerce. Most of the systems are based on the use of English language. The incorporation of multilingual capacities in companies supposes an effective increasing of market and also of image. The advantage is that the amount of work to encode any text that belongs to a web-site is the same disregarding the number of source languages. It is only needed the target language generator (at the moment there exist more than ten languages generators covering the 85% of the human population). The integration of the systems is very easy and has not special complications.

However, where the UNL system should have more impact is on the B2B activities. All the concepts and components from the B2B, but particularly the ontologies based on cXML [Merkov, 1999], to define business documents defining technical and business dictionaries (completely compatible with the UW dictionaries). OBI's [OBI, 1999] data formats rely on EDI standards for document exchanges etc. which are completely compatible with the structure of the UNL Documents. The international commercial exchange and current growth of the E-commerce are due to this kind of exchanges. Here the availability of multilingual systems able to support the exchange of documents, transactional information, a correct common understanding of contractual documentation (well addressed by a common UNL codification) is perhaps the most important application of this system. In addition, corporate information and even more complex systems (as multilingual e-mail) can be supported.

From Bilingual to Multilingual Translation Systems

The UNL system could be viewed as an alternative to the classical machine translations systems. However, it is not exactly the case. When the classical machine translation systems massively follow the model of “transfer”, the UNL is conceived in a different way. First of all, the UNL system is not a system to support machine translation but multilingual services. It is not the same. There is not any automatic conversion from a language to UNL.

For instance, analysers and dictionaries of a particular language can be integrated with the production of UNL code at the required level. An existing dictionary can be reused to develop the UW and thus to develop the UNL Dictionary for a language or a specific domain. The target language generators of an existing language can be reused once integrated the input with the UNL code. These operations permit the transformation of a bilingual system into a multilingual one. In fact, the Russian Language Centre* [http://www.unl.ru], the French Language Centre and some others are sustaining the UNL system reusing bilingual pre-existing machine translation systems. Thus, this means that there are two types of users of this system, the industry itself manufacturing the integration and therefore creating multilingual systems with a high degree of reuse of the linguistic repository and the final users of the machine translation systems like human translators, that are normally in charge of the post-editing of the target documents. The main advantage is that these persons will increase their productivity because while working in just one language, they are producing contents in many other languages.

References

[Bateman, 1995] Bateman, J.A; Henschel, R. and Rinaldi, F. The Generalized Upper Model 2.0. 1995. Available on line at http:// www.darmstadt.gmd.de/publish/komet/gen-um/newUM.html

[Cardeñosa, 2004] Cardeñosa, J., Gallardo, C., and Iraola. “The forgotten key point for assuring knowledge consistency in CLIR systems”. In: Proceedings of the Workshop Lessons Learned from Evaluation: Towards Transparency & Integration in Cross-Lingual Information Retrieval (LREC, 2004), Lisbon, May, 2004.

[Cardeñosa, 2005]. Cardeñosa, J., Gallardo, C., and Iraola, L. An XML-UNL Model for Knowledge-Based Annotation. Research on Computing Science, vol 12, pp 300-308. ISBN: 970-36-0226-6. México, 2005

[CyC] www.cyc.com

[Fellbaum, 1998] Fellbaum, C., editor. WordNet: An Electronic Lexical Database. Language, Speech, and Communication Series. MIT Press, 1998

[Hailaoui, 2005]. Hailaoui, N. and Boitet, C. A Pivot XML-Based Architecture for Multilingual, Multiversion Documents: parallel monolingual documents aligned through a central correspondence descriptor and possible use of UNL. Research on Computing Science, vol 12, pp 309-326. ISBN: 970-36-0226-6. México, 2005

[Merkov, 1999] Merkov, M. cXML: A new Taxonomy for E.-Commerce. 1999. Available on line at http://ecommerce.internet.com/outlook/article/0,1467,7761_124921,00.html

[Muraki, 1989] Muraki, K. PIVOT: Two-phase machine translation system. Proceedings of the Second Machine Translation Summit, Tokyo, 1989

[Nicholas, 2000] Nicholas, L.; and Lockwood, R. Final Report: SPICE-PREPII: Export potential and linguistic customization of digital products and services. EPS Ltd and Equipe Consortium Ltd, 2000

[OBI, 1999] The Open Buying on the Internet (OBI) Consortium. 1999. Open buying on the Internet. OBI specification. 2000. Available on line at http://www.openbuy.org/

[Schank, 1972] Schank, R.C. Conceptual Dependency: A Theory of Natural Language Understanding, Cognitive Psychology, Vol 3, 532-631, 1972

[Sowa, 1996] Sowa, J.F. "Processes and participants," in Eklund et al., eds. (1996) Conceptual Structures: Knowledge Representation as Interlingua, Lecture Notes in AI -1115, Springer-Verlag, Berlin, pp. 1-22, 1996

[Uchida, 1989] Uchida, H. ATLAS-II: A machine translation system using conceptual structure as an Interlingua. Proceedings of the Second Machine Translation Summit, Toky, 1989

[Uchida, 2003] Uchida, H. The Universal Networking Language. Specifications. 2003. Available on-line at http://www.undl.org

Authors' Information

Jesús Cardeñosa – UPM.Artificial Intelligence Department, Universidad Politécnica de Madrid, Campus de Montegancedo, s/n. 28660 Madrid, SPAIN; email: carde@opera.dia.fi.upm.es

Carolina Gallardo – UPM. Artificial Intelligence Department, Universidad Politécnica de Madrid, Campus de Montegancedo, s/n. 28660 Madrid, SPAIN; email: carolina@opera.dia.fi.upm.es

Eugenio Santos – UPM. Business organisation Department. Universidad Politécnica de Madrid. Carretera de Valencia Km.7. 28041 Madrid, SPAIN; email: esantos@eui.upm.es

* Language Centres are the operational national units that support the development of specific local language within the worldwide organization of the UNL Programme.

A NEW APPROACH FOR ELIMINATING THE SPURIOUS STATES IN RECURRENT NEURAL NETWORKS

Víctor Giménez-Martínez, Carmen Torres,
José Joaquín Erviti Anaut, Mercedes Perez-Castellanos

Abstract: As is well known, the Convergence Theorem for the Recurrent Neural Networks, is based in Lyapunov´s second method, which states that associated to any one given net state, there always exist a real number, in other words an element of the one dimensional Euclidean Space R, in such a way that when the state of the net changes then its associated real number decreases. In this paper we will introduce the two dimensional Euclidean space R^2*, as the space associated to the net, and we will define a pair of real numbers* (x, y)*, associated to any one given state of the net. We will prove that when the net change its state, then the product* $x \cdot y$ *will decrease. All the states whose projection over the energy field are placed on the same hyperbolic surface, will be considered as points with the same energy level. On the other hand we will prove that if the states are classified attended to their distances to the zero vector, only one pattern in each one of the different classes may be at the same energy level. The retrieving procedure is analyzed trough the projection of the states on that plane. The geometrical properties of the synaptic matrix* W *may be used for classifying the n-dimensional state-vector space in n classes. A pattern to be recognized is seen as a point belonging to one of these classes, and depending on the class the pattern to be retrieved belongs, different weight parameters are used. The capacity of the net is improved and the spurious states are reduced. In order to clarify and corroborate the theoretical results, together with the formal theory, an application is presented*

Keywords: Learning Systems, Pattern Recognition, Graph Theory, Image Processing, Recurrent Neural Networks.

ACM Classification Keywords: I.2.6 Learning: Connectionism and neural nets; G.2.2. Graph Theory; I.4.0 Image processing software

1. Introduction

The problem to be considered when Recurrent Neural Networks (RNN) are going to be used as *Pattern Recognition* systems, is how to impose prescribed prototype vectors $\xi^1, \xi^2, ..., \xi^p$, of the space $\{-1,1\}^n$, as fixed points. In the classical approach, the synaptic matrix $W = (w_{ij})$ should be interpreted as a sort of sign correlation matrix of the prototypes. The element $w_{ij} \in W$, is going to represent some kind of relation between coincidences and not coincidences on the list of the components "i" and "j" for all the prototype vectors $\xi^1, \xi^2, ..., \xi^p$. The classical solution to impose fixed points by means of the synaptic matrix W is the *Hebb´s* law, which states that the synaptic weight w_{ij} should increase whenever neurons "i" and "j" have simultaneously the same activity level and it should decrease in the opposite case. As it was pointed out above, the prototype vector components must belong to the set $\{-1,1\}$; this fact is the cornerstone of the *Hebb´s* law mathematical interpretation. The reason is that when the prototype ξ^μ is stored, neurons "i" and "j" may receive a similar sign or not". The mathematical advantage of this interpretation lies in the fact that when the prototype ξ^μ is acquired, the synaptic weight w_{ij} should increase if neurons "i" and "j" receive a similar sign: in other words if $\xi_i^\mu . \xi_j^\mu$ is positive. On the other hand w_{ij} should decrease if $\xi_i^\mu . \xi_j^\mu$ is negative. The updating of the weights may be then expressed by, $\Delta w_{ij} = \xi_i^\mu . \xi_j^\mu$, in other words, when the sign of the components "i" and "j" in the prototype ξ^{μ} are with similar sign, the weight w_{ij} is positively reinforce, otherwise do the same but in a negative sense. In general a positive learning parameter η may be used, and it can be state as the general training rule that the prototype ξ^{μ} is stored then $\Delta w_{ij} = \eta . \xi_i^{\mu} . \xi_j^{\mu}$ (being η a positive learning factor); which means that, $\Delta w_{ij} \in \{-\eta, \eta\}$. The synaptic matrix W should be interpreted as a sort of sign correlation matrix of the prototypes.

2. Training: Parameters of the Net

In our approach, instead of a matrix for storing the weights, a weight vector $\vec{p} = (p_1, p_2, ..., p_n)$ is going to be introduced. At the beginning $\vec{p} = (0,0,...,0)$. At time t, when the training pattern ξ^{μ} is acquired, the weight vector $\vec{p}$, will be updated by this very simple rule:

$$\vec{p} = \vec{p} + \xi^{\mu}$$

which means that, $\forall i, j = 1,..,n$ then ,

$$\begin{cases} \text{If } \left(\xi_i^{\mu}=1 \text{ and } \xi_j^{\mu}=1\right) \text{ then } \left(p_i = p_i + 1 \text{ and } p_j = p_j + 1\right) \\ \text{If } \left(\xi_i^{\mu}=-1 \text{ and } \xi_j^{\mu}=-1\right) \text{ then } \left(p_i = p_i - 1 \text{ and } p_j = p_j - 1\right) \\ \text{If } \left(\xi_i^{\mu}=-1 \text{ and } \xi_j^{\mu}=1\right) \text{ then } \left(p_i = p_i - 1 \text{ and } p_j = p_j + 1\right) \\ \text{If } \left(\xi_i^{\mu}=1 \text{ and } \xi_j^{\mu}=-1\right) \text{ then } \left(p_i = p_i + 1 \text{ and } p_j = p_j - 1\right) \end{cases}$$

It is clear that in this way training is faster than in the classical procedure, and the knowledge stored in the net parameter is equivalent. The synaptic matrix $W = (w_{ij})$, may be rebuilt by doing,

$$w_{ij} = p_i + p_j, \quad \forall i, j = 1,..,n$$

Which is equivalent to state that when the prototype ξ^{μ} is acquired, the synaptic weight w_{ij} should increase if neurons "i" and "j" are in state 1, and will decrease if neurons "i" and "j" are in state -1. We may then consider that with this procedure, instead of storing some kind of sign correlation of the prototypes, like in the classical procedure was done, we are storing the correlation prototypes features. This method has also a very good property, and this is that, for any $i, j, r, s \in \{1,..,n\}$, then as,

$$p_i + p_j + p_r + p_s = p_i + p_s + p_r + p_j$$

one has that

$$w_{ij} + w_{rs} = w_{is} + w_{rj}$$

So, as the way the parameters are stores is related with the features of the pattern components, we are going to use in our approach the Boolean space $\{0,1\}^n$ instead of the bimodal space $\{-1,1\}^n$. The training procedure will be then defines by

$$\Delta w_{ij} = \begin{cases} +1 & \text{if } \xi_i^{\mu} = \xi_j^{\mu} = 1, i \neq j, \\ -1 & \text{if } \xi_i^{\mu} = \xi_j^{\mu} = 0, i \neq j, \\ 0 & \text{otherwise.} \end{cases}$$

Two important advantages may be extracted from the above approach. The first advantage is that the space required for storing the parameters is lesser than in the classical one: the parameters may be stored in the weight vector, and then doing $w_{ij} = p_i + p_j$, to span it to the weight matrix if necessary. The second advantage is that the training may be much more easily understood using the next graphical interpretation of the training algorithm: A n complete graph G may be introduced associated with the net. At the first step, a null value is assigned to all the edges a_{ij}, then, when a learning pattern ξ^{μ} is acquired by the net, it is superposed over the graph G. The components, $\{\xi_1^{\mu},...,\xi_n^{\mu}\}$, are going to be mapped over the vertices $\{v_1,...,v_n\}$ of G. This mapping may be interpreted as a coloring of the edges in G, in such a way that, if $\xi_i^{\mu} = \xi_j^{\mu} = 1$, the edge a_{ij} (whose ending vertices are v_i and v_j) will be colored with a certain color, for example red. On the other hand, if $\xi_i^{\mu} = \xi_j^{\mu} = 0$, then a_{ij} will be colored with a different color, as for example blue. The rest of the edges in G remain uncolored. Once this coloring has been done, the value assigned over the, also complete, graph of red edges are positively reinforced and the value assigned over the edges of the blue graph are negatively reinforced. The value over the rest of the edges remains unchanged. Once the pattern ξ^{μ} is acquired, the colors are erased and we repeat the same color assignation with the next pattern to be acquired by the net, and so on. When every vector in the training pattern set has been integrated in the net, the training stage is finished, the resulting graph G has become edge-valued and its weight matrix is the synaptic matrix $W = \left(w_{ij}\right)$ of the net. Now if we define the basic matrix $U^k = (u_{ij}^k)$, where

$$\begin{cases} u_{ij} = 1 & \text{if } \left(i = k \ \text{ xor } \ j = k\right) \\ u_{ij} = 0 & \text{otherwise} \end{cases}$$

then, any synaptic matrix W is generated by the set of basic matrices $\left\{U^1, U^2, ..., U^n\right\}$. In other words,

$$W = p_1.U^1 + ... + p_n.U^n .$$

3. Recall

For recalling a pattern from the net, the net should be colored with the color associated with that pattern, which may be interpreted as if the net had in a certain state $x\left(t\right)$. Then, it is clear that, we may define the energy pair number $\left\{I\left(t\right), O\left(t\right)\right\}$, where

$$I\left(t\right) = \frac{1}{2}\left(x(t)\ .\ W\ .\ x(t)^t\right)$$

represents the sum of the values of all the parameters w_{ij}, associated with all the edges colored in red, and if $\overline{x}\left(t\right)$ is the symmetric vector of $x\left(t\right)$, then

$$O\left(t\right) = \frac{1}{2}\left(\overline{x}(t)\ .\ W\ .\ \overline{x}(t)^t\right)$$

represents the sum of the values of all the parameters w_{ij}, associated with all the edges colored in blue. Taking now into account that

$$W = p_1 \cdot U^1 + ... + p_n \cdot U^n$$

one has that

$$\frac{1}{2}\left(x(t)\ \cdot\ W\ \cdot x(t)^t\right) = \frac{1}{2}x(t)(p_1U^1 + .. + p_nU^n)x(t)^t = p_1\left[\frac{1}{2}x(t)\cdot U^1 \cdot x(t)^t\right] + ... + p_1\left[\frac{1}{2}x(t)\cdot U^n \cdot x(t)^t\right]$$

and

$$\frac{1}{2}\left(\overline{x}(t)\ \cdot\ W\ \cdot\ \overline{x}(t)^t\right) = \frac{1}{2}\overline{x}(t)(p_1U^1 + .. + p_nU^n)\overline{x(t)}^t = p_1\left[\frac{1}{2}\overline{x}(t)\cdot U^1 \cdot \overline{x}(t)^t\right] + .. + p_1\left[\frac{1}{2}\overline{x}(t)\cdot U^n \cdot \overline{x}(t)^t\right]$$

In other words, if $\forall i, j = 1,..,n$, we do

$$\begin{cases} I_i = \frac{1}{2} x(t) \cdot U^i \cdot x(t)^t \\ O_i = \frac{1}{2} \bar{x}(t) \cdot U^i \cdot \bar{x}(t)^t \end{cases}$$

then

$$\begin{cases} I(t) = p_1 \cdot I_1(t) + ... + p_n \cdot I_n(t) \\ \\ O(t) = p_1 \cdot O_1(t) + ... + p_n \cdot O_n(t) \end{cases}$$

but if n_1 is the number of unit components of $x(t)$ and if n_0 is the number of the null ones (in other words n_1 is the Hamming distance from $x(t)$ to the zero vector), it is obvious that

$$I_i(t) = \begin{cases} n_1 - 1 & if \ \ x_i(t) = 1 \\ 0 & if \ \ x_i(t) = 0 \end{cases}, \quad \text{and} \quad O_i(t) = \begin{cases} n_0 - 1 & if \ \ x_i(t) = 0 \\ 0 & if \ \ x_i(t) = 1 \end{cases}$$

So, if $i_1, i_2, ..., i_{n_1}$, and $j_1, j_2, ..., j_{n_0}$ are the places where the unit and null components of $x(t)$, are respectively located, the above equations could be written as

$$I(t) = (n_1 - 1)(p_{i_1} + p_{i_2} + ... + p_{i_{n_1}}) \text{ and } O(t) = (n_0 - 1)(p_{j_1} + p_{j_2} + ... + p_{j_{no}})$$

which means that

$$\frac{I(t)}{n_1 - 1} + \frac{O(t)}{n_0 - 1} = k \text{ (where } k = p_1 + ... + p_n \text{).}$$

In other words: all states $x(t)$ sharing the same distance "j" (where $j \in \{1,2,...,n\}$), will verify the before equation. The states' space could be then classified in the n classes $[1],[2],...,[j],...,[n]$. If the pair $\{I(t), O(t)\}$, where defined as the energy pair (PE), associated to $x(t)$, one could state that all the states in the same class, has theirs PE´s in the same energy line. On the other hand, we may define the relative weight of the neuron "i" when the net is in state $x(t)$, as the contribution of this neuron to the component $I(t)$, if $x_i(t) = 1$; or as the contribution of this neuron to the component $O(t)$, if $x_i(t) = 0$. So, if $x_i(t) = 1$, we define the *relative weight* $w_i(t)$ of the neuron "i" when the net is in state x(t) as:

$$w_i(t) = \frac{\sum_{j=1}^{n} x_j(t) \cdot w_{ij}}{I(t)}$$

and taking into account that,

$$\sum_{j=1}^{n} x_j(t).w_{ij} = x_1(t)(p_i + p_1) + .. + x_i(t) \ \cdot \ 0 + .. + x_n(t)(p_i + p_n) =$$

$$(x_1(t) \cdot p_1 + ... + x_n(t) \cdot p_n) + p_i (x_1(t) + ... + x_n(t)) - 2 \cdot x_i(t).p_i = p \ \cdot \ x(t) + p_i(n_1 - 2)$$

and

$$I(t) = (n_1 - 1) \cdot p.x(t)$$

$w_i(t)$ may be represented as

$$w_i(t) = \frac{1}{n_1 - 1} + \frac{n_1 - 2}{n_1 - 1} \cdot \frac{p_i}{p \cdot x(t)}$$

and without difficult it could be also proved that if $x_i(t) = 0$, then

$$w_i(t) = \frac{1}{n_0 - 1} + \frac{n_0 - 2}{n_0 - 1} . \frac{p_i}{p.\bar{x}(t)}$$

4. Dynamic Equation

If in time t the state vector $x(t)$ is in class $[j]$, then for any i from 1 to n, the dynamic equation is defined as

$$x_i(t+1) = f_h\left[\left(f_b(x_i(t))\right).\left(w_i(t) - \theta_j\right)\right]$$

where f_h is the Heaviside step function and f_b is the function defined as $f_b(x) = 2 \cdot x - 1$, which achieves the transformation from the domain $\{0,1\}$ to the domain $\{-1,1\}$. In other words: if $x(t)$ is in class $[j]$ and $x_i(t) = 1$ the above expression could be written as

$$x_i(t+1) = f_h\left(w_i(t) - \theta_j\right)$$

which states that if $w_i(t) < \theta_j$, then $x_i(t)$ changes its state from state $x_i(t) = 1$ to $x_i(t) = 0$; and otherwise $x_i(t)$ doesn't change its state. On the other hand, if $x(t)$ is in class $[j]$ and $x_i(t) = 0$, the above expression could be written as

$$x_i(t+1) = f_h\left(\theta_j - w_i(t)\right)$$

which states that if $w_i(t) < \theta_j$, then $x_i(t)$changes its state from state $x_i(t) = 0$ to $x_i(t) = 1$; and otherwise $x_i(t)$ doesn't change its state. The value of θ_j is the j-th component of a n-dimensional_vector $\bar{\theta}$ and is related with the class $[j]$ to which the state $x(t)$ belongs, and must not to be interpreted as a threshold for the "i" unit (which will be assumed to be zero, whenever this hypothesis is not critical for the results we will to establish). It is clear that the lower we set the value of θ_j, the more states in class $[j]$ will have theirs relatives weights greater than θ_j which means that more fixed points the class $[j]$ will have. The desirable values for the, so to be called, capacity vector parameter $\bar{\theta} = (\theta_1, ..., \theta_n)$, may be obtained in an adaptive way. It can also be stated that the sum of the relative weights $w_i(t)$ for the unit components of $x(t)$ is equal to 2. The same could be proved for the null components. We have then that the relative weight vector $w(t) = (w_1(t), w_2(t), ..., w_n(t))$ associated to any state vector $x(t)$ may also be interpreted as a sort of frequency distribution of probabilities. The reason is that

$$\sum_{i=1}^{n} w_i(t) = 4 \;\Rightarrow\; \sum_{i=1}^{n} \frac{1}{4} \cdot w_i(t) = 1$$

For any relative weight vector $w(t)$. The "uniform distribution vector" would be the one with all its components equal to $4/n$, which mean that the relative weight vector may be interpreted as a sort of frequency distribution of probabilities, this distribution may be considered as the relative weight vector associated to that state.

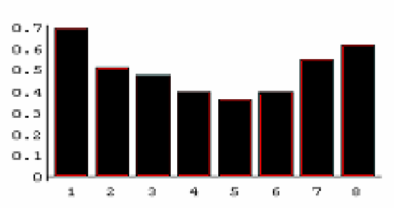


Figure 1 Relative Weight Vector associated to a certain state x, in a space of dimension 8.

5. Application

Our algorithm has been used in several applications. In this paper, we take, as an example for validating the performance of the algorithm we propose, the problem of the recognition of the Arabian digits as the prototype vectors:

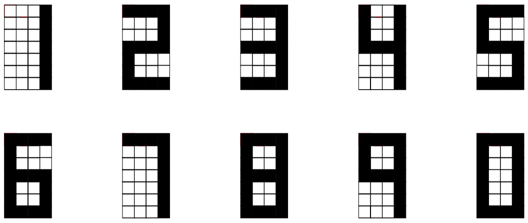


Figure 2 Arabian digits

Where the dimension n, of the pattern space is 28, and ξ^1=[0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1], ξ^2 = [1,1,1,1,0,0,0,1,0,0,0,1,1,1,1,1,1,0,0,0,1,0,0,0,1,1,1,1], and so on. After training, the weight vector is p = 1/14 {53, 25, 25, 53, 11, -73, -73, 39, 11, -73, -73, 39, 39, 25, 25, 67, -17, -73, -73, 53, -17, -73, -73, 53, 11, 11, 11, 67}.

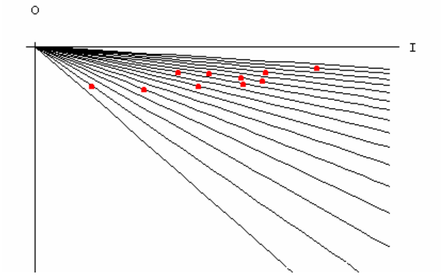


Figure 3 Arabian Digits Projections

In figure 3 the reader may see the energy lines and theirs associated PE´s. The Arabian digits are in this way placed on the lines: r_7, r_{16}, r_{16}, r_{13}, r_{16}, r_{15}, r_{10}, r_{20}, r_{15}, r_{18}. And the associated PE´s are 1/7{1113,-3710}, 1/7{3420,-2508}, 1/7{4470,-3278}, 1/7{3210,-3745}, 1/7{4050,-2970}, 1/7{2821,-2418}, 1/7{2133,-4029}, 1/7{5548,-2044}, 1/7{4095,-3510}, 1/7{4539,-2403}. The problem now is how to obtain in an adaptive way the capacity parameters $\theta_1,\theta_2,\ldots,\theta_{28}$, in order to obtain the Arabian digits as fixed points with the least number of parasitic points as possible. When the before dynamic equation is considered, a point $x(t)$ whose energy projection belongs to the r_j line, is a fixed point if, and only if, the (capacity) parameter θ_j is an upper bound for all the relative weights $w_i(t)$ associated to the components of $x(t)$.

Once the training has finished, the relative weight vector of the prototypes could then be calculated using. If the energy projection of the prototype ξ^μ belongs to r_j and the largest of the components of $w_i(t)$ is taken as θ_j: it is clear that the prototype ξ^μ will be a fixed point. But the problem is how to avoid that a point with high degree of correlation with a prototype but with all its relative weights components lower than the capacity parameter to skip away from this prototype. In our example, the number 2 is a prototype with 16 units components, in other words its energy projection is placed in the line r_{16} and the capacity parameter θ_{16} is equal to 0.0741306. If a little noise is added to the pattern (pattern in figure 4) and this noisily pattern (belonging to class r_{15}) is given for retrieving:

It may happen that the noisily pattern changes to other state quite different from its natural attractor (the number 2) The reason for that, is that the prototype number 6, see figure 5, belongs also to the class r_{15} and the stability condition in this class was set very high.

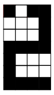


Figure 4 Noisily Prototype 2 belonging to r_{15}

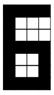


Figure 5 Prototype 6 belonging to r_{15}

The question is how to get that only the second component changes its state, when the Noisily Prototype 2, is given for retrieving. In other words, how to get all the neighbors (inside a give radius) of a prototype to be attracted by this prototype, see figure 6. The idea, proposed in this paper, made use of the deviation defined in [Giménez 2000]. When, in time t, the dynamic equation is applied to a component of the vector $x(t)$, this component will change its state not only if the relative weight $w_i(t)$ is lower that the capacity parameter of its class. The deviation of the new state, in the case of change of sate, must be similar to the deviation of the prototypes in the new class. The degree of similarity may be measured by a coefficient μ. The coefficient μ is handled in a dynamical way (the more is the time the higher is the coefficient).

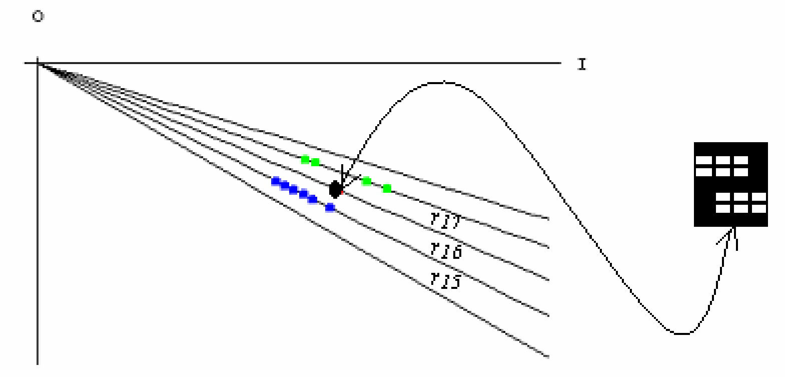


Figure 6 Neighbor of prototype 2 in r_{15} and r_{17}

Besides the weight vector, there is other set of parameters of the net. For every one class r_i, the capacity parameter θ_l and the deviation of the prototypes in this class are obtained. So the algorithm control not only if the new state is strongly correlate with some prototype in its class, the algorithm also control that the components in the new state must, with a high degree of probability, be placed in similar places as some prototype of the class. We have applied with to our example, obtaining that almost all the points inside a neighborhood of radius 1, of the prototypes, are attracted by these prototypes. The 10 Arabian digits are fixed points of the system, and almost all the 28 neighbor of any one of them were attracted by its attractor prototype. In figure 7, the number of points inside a neighborhood of radius 1, of the prototypes are expressed.

24	→	1	22	→	6
23	→	2	25	→	7
25	→	3	25	→	8
22	→	4	22	→	9
27	→	5	21	→	0

Figure 7 Prototype 6 belonging also to r_{15}

6. Conclusion

The weight parameters in the Hopfield network are not a free set of variables. They must fulfill a set of constrains which have been deduced trough a new re-interpretation of the net as Graph Formalisms. Making use of this constrains the state-vector has been classified in n classes according to the n different possible distances from any of the state-vectors to the zero vector. The $(n \times n)$ matrix of weights may also be reduced to an n-vector of weights. In this way the computational time and the memory space, required for obtaining the weights, is optimized and simplified. The degree of correlation from a pattern with the prototypes may be controlled by the dynamical value of two parameters: the capacity parameter θ which is used for controlling the capacity of the net (it may be proved that the bigger is the θ_j component of θ, the lower is the number of fixed points located in the r_j energy line) and the parameter μ which measures the deviation to the prototypes. A typical example has been exposed; the obtained results have proved to improve the obtained when the classical algorithm is applied.

Bibliography

[Hopfield 82.] J. J. Hopfield. Neural Networks and physical systems with emergent collective behavior. Proc. Natl. Acad. Sci. USA, 79:2554, 1982.

[Kinzel 85] W. Kinzel, Z. Phis, (B-Condensed Matter) Learning and pattern recognition in spin glass models, vol.60, pp.205-213, 1985

[Elice 87] R. Mc. Elice, E. Posner, E. Rodemich, and S. Venkatesh. The capacity of the Hopfield associative memory. IEEE Trans. On Information Theory, vol.IT-33. pp.461-482, 1987.

[Bose 96] N. K. Bose and P. Liang. Neural Network Fundamentals with Graphs, algorithms and Applications. McGraw Series in Electrical and Computer Engineering.1996.

[Giménez 97] V. Giménez-Martínez, M. Pérez-Castellanos, J. Ríos Carrión and F. de Mingo, Capacity and Parasitic Fixed points Control in a Recursive Neural Network, Lecture Notes in Computer Science, SPRINGER-VERLAG, 1997, pp.215-226

[Giménez 2000] V. Giménez-Martínez. A Modified Hopfield Algorithm Auto-Associative memory with Improved Capacity, IEEE Transactions on Neural Networks, vol.11,n.4, 2000, pp. 867-878.

[Giménez 2001] Giménez-Martínez V., Erviti Anaud J. and Pérez-Castellanos M.M, Recurrent Neural Networks for Statistical Pattern Recognition, Frontiers in Artificial Intelligence and Applications, Vol.69, part 1, n.3, 2001, pp.1152-1159.

[Giménez 2001] Giménez-Martínez V., Aslanyan L., Castellanos J.and Ryazanov V., Distribution functions as attractors for Recurrent Neural Networks Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications, vol.11, n.3, 2001, pp.492-497.

Authors' Information

V. Giménez-Martínez – Dep. de Matemática Aplicada.

C. Torres – Dep. de Matemática Aplicada.

J. Erviti Anaut – Dep. de Matemática Aplicada.

M. Perez-Castellanos – Dep de Arquitectura y Tecnología. de Sistemas

Facultad de Informática, U.P.M. Campus de Montegancedo s/n Boadilla del Monte, 28660 MADRID, SPAIN

e-mail: vgimenez@fi.upm.es

A GEOMETRICAL INTERPRETATION TO DEFINE CONTRADICTION DEGREES BETWEEN TWO FUZZY SETS

Carmen Torres, Elena Castiñeira, Susana Cubillo, Victoria Zarzosa

Abstract: *For inference purposes in both classical and fuzzy logic, neither the information itself should be contradictory, nor should any of the items of available information contradict each other. In order to avoid these troubles in fuzzy logic, a study about contradiction was initiated by Trillas et al. in [5] and [6]. They introduced the concepts of both self-contradictory fuzzy set and contradiction between two fuzzy sets. Moreover, the need to study not only contradiction but also the degree of such contradiction is pointed out in [1] and [2], suggesting some measures for this purpose. Nevertheless, contradiction could have been measured in some other way. This paper focuses on the study of contradiction between two fuzzy sets dealing with the problem from a geometrical point of view that allow us to find out new ways to measure the contradiction degree. To do this, the two fuzzy sets are interpreted as a subset of the unit square, and the so called contradiction region is determined. Specially we tackle the case in which both sets represent a curve in $[0,1]^2$. This new geometrical approach allows us to obtain different functions to measure contradiction throughout distances. Moreover, some properties of these contradiction measure functions are established and, in some particular case, the relations among these different functions are obtained.*

Keywords: *fuzzy sets, t-norm, t-conorm, fuzzy strong negations, contradiction, measures of contradiction.*

ACM Classification Keywords: *F.4.1 Mathematical Logic and Formal Languages: Mathematical Logic (Model theory, Set theory); I.2.3 Artificial Intelligence: Deduction and Theorem Proving (Uncertainty, "fuzzy" and probabilistic reasoning); I.2.4 Artificial Intelligence: Knowledge Representation Formalisms and Methods (Predicate logic, Representation languages).*

Introduction

One of the main problems tackled by fuzzy logic is how to deal with inferences that include imprecise information. So, several methods have been proposed within this field for inferring new knowledge from the original premises. In any inference process, however, we have to assure that the results yielded neither contradict each other nor the original information.

The concept of contradiction in fuzzy logic was introduced by Trillas *et al.* in [5] and [6]. These papers formalize the idea that a fuzzy set *P* associated with a vague predicate **P** is contradictory if it violates the principle of non-contradiction in the following sense: the statement "If *x* is **P**, then *x* is **not P**" holds with some degree of truth. So, they established that the fuzzy set P is contradictory regarding an involutive negation N if μ_P (x) ≤ $(N\circ\mu_P)$ (x) for all x, where $\mu_P(x)$, named the membership function of *P*, represents the degree in which *x* satisfies the predicate **P**. Contradiction between two fuzzy sets was also introduced in [5] and [6]. Analogously, two fuzzy sets *P* and *Q* are N-contradictory if the condition μ_P (x) ≤ $(N\circ\mu_Q)$ (x) holds for all x. The need to speak not only of contradiction but also of degrees of contradiction was later raised in [1] and [2], where a function was considered for the purpose of determining (or measuring) the contradiction degree of a fuzzy set. Also, in [2] the authors proposed a function that appears to be suited for measuring the degree of contradiction between two fuzzy sets. However, many functions could be constructed for these purposes, and it is useful to specify what conditions a function must meet to be used as a measure of contradiction. Specifically, some axioms are needed to be able to decide whether a function is suitable for measuring the degree of contradiction. These axioms were established in [3].

In this work, we retake the study of the contradiction between two fuzzy sets, focusing on the problem from a geometrical perspective that suggests new ways of defining measures of contradiction. Therefore, after a geometrical study to determine what we will name regions of contradiction and non-contradiction, we will then define some functions by analyzing some of its properties.

Preliminaries

Firstly, we will introduce a series of definitions and properties for their subsequent development in this article.

Definition 2.1 ([7]) A fuzzy set (FS) *P*, in the universe $X \neq \emptyset$, is a set given as $P=\{(x,\mu(x)): x \in X\}$ such that, for all $x \in X$, $\mu(x) \in [0,1]$, and where the function $\mu: X \to [0,1]$ is called membership function. We denote $\mathcal{F}(X)$ the set of all fuzzy sets on X.

Definition 2.2 $P \in \mathcal{F}(X)$ with membership function $\mu \in [0,1]^X$ is to be said a normal fuzzy set if $\mathrm{Sup}\{\mu(x) : x \in X\}=1$.

Definition 2.3 A fuzzy negation (FN) is a non-increasing function $N: [0,1] \to [0,1]$ with $N(0)=1$ and $N(1)=0$. Moreover, N is a strong fuzzy negation if the equality $N(N(y))=y$ holds for all $y \in [0,1]$.

The strong negations were characterized by Trillas in [4]. He showed that N is a strong negation if and only if, there is an order automorphism g in the unit interval (that is, $g:[0,1] \to [0,1]$ is an increasing continuous function with $g(0)=0$ and $g(1)=1$) such that $N(y)=g^{-1}(1-g(y))$, for all $y \in [0,1]$; from now on, let us denote $N_g=g^{-1}(1-g)$. Furthermore, the only fixed point of N_g is $n_g=g^{-1}(1/2)$.

Measuring N_g-contradiction between Two Fuzzy Sets

As mentioned above, μ and σ are said to be N_g-contradictory if $\mu(x) \leq N_g(\sigma(x))$ for all elements x in the universe of discourse, which is equivalent to $\mu(x) \leq g^{-1}(1-g(\sigma(x)))$ for all x, and also to $\mathrm{Sup}\{g(\mu(x))+g(\sigma(x)) / x \in X\} \leq 1$. Here again ascertaining whether two sets are contradictory will fall short of the mark, and a distinction should be made between any differing degrees of contradiction occurring in such situations. This problem was addressed for the first time in [1] and [2].

In this section, in order to study the degree of N_g-contradiction between two fuzzy sets *P* and *Q* (with membership functions $\mu, \sigma \in [0,1]^X$, respectively) we consider the set $\{(\mu(x),\sigma(x)) : x \in X\}$ as a subset of $[0,1]^2$ (we denote it by $X_{\mu\sigma}$ to be short) and we will firstly analyze in what regions of $[0,1]^2$ $X_{\mu\sigma}$ must remain provided that μ and σ are N_g-contradictory (see figure 1). The aim of this analysis is to find some relation suggesting the way of measuring the N_g-contradiction between two fuzzy sets. Secondly, we propose some possible functions in order to measure the degrees of contradiction, bearing in mind the mentioned analysis.

Regions of N_g-contradiction

As mentioned above, given $\mu, \sigma \in [0,1]^X$ and a strong negation N_g, then μ and σ are N_g-contradictory if and only if

$$\mu(x) \leq N_g(\sigma(x)) \ \forall x \in X \Leftrightarrow \sigma(x) \leq N_g(\mu(x)) \ \forall x \in X \Leftrightarrow g(\mu(x))+g(\sigma(x)) \leq 1 \ \forall x \in X$$

The above inequalities determine a curve in the unit square, with equation $y_1=N_g(y_2)$ or $y_2=N_g(y_1)$ or $g(y_1)+g(y_2)=1$; this curve, called the *limit curve of N_g-contradiction,* is the border between two regions: the region in which contradictory sets remain and the region free of contradiction (see figure 1).

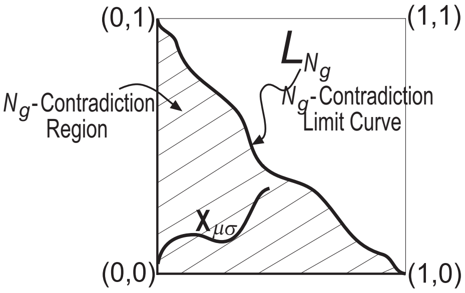


Figure 1: N_g-contradiction region and N_g-contradiction limit curve

Let us see these regions in several particular cases and after that, the general case will be discussed.

(a) N_s-contradiction with standard negation $N_s(y)=1-y$

Let $N_s=1\text{-}id$ be the standard negation that is generated by g=id. Then μ and σ are N_s-contradictory if and only if $\mu(x)+\sigma(x) \leq 1$ for all $x \in X$, that is equivalent to $X_{\mu\sigma} \subset \{(y_1,y_2) \in [0,1]^2 : y_1+y_2 \leq 1\}$ (see figure 2(a)).

(b) N_g-contradiction with g(y)=y^2

The order automorphism g(y) = y^2 determines the strong negation $N_g(y) = \sqrt{1-y^2}$, and the sets μ, σ ∈ $[0,1]^X$ are N_g-contradictory if and only if $\mu(x) \le \sqrt{1-\sigma(x)^2}$, that is equivalent to $\mu(x)^2 + \sigma(x)^2 \le 1$. Therefore, μ and σ are N_g-contradictory if and only if

$$X_{\mu\sigma} = \{(\mu(x),\sigma(x)) : x \in X\} \subset \{(y_1,y_2) \in [0,1]^2 : y_1^2 + y_2^2 \le 1\}$$

Then, $X_{\mu\sigma}$ must remain inside or on the circumference with center (0,0) and radius 1 (see figure 2 (b)).

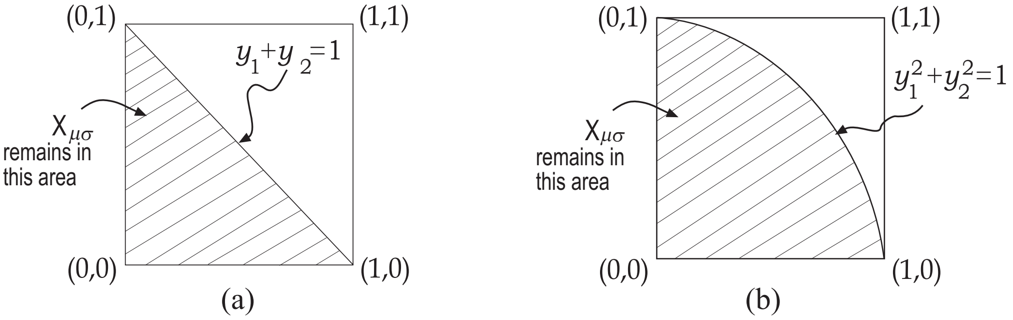


Figure 2: (a) N_s-contradiction area and (b) N_g-contradiction area with g(y)=y^2

(c) N_r-contradiction with N_r determined by g(y)=y^r, r>0

Let's consider the family of strong negations $\{N_r\}_{r>0}$, where for each r>0 N_r is determined by the automorphism $g_r(y)=y^r$. This family includes as particular cases the negations given in (a) and (b) and for each r>0 is $N_r(y)=(1-y^r)^{1/r}$ with a fixed point $y_{N_r} = \dfrac{1}{2^{1/r}}$. μ, σ ∈ $[0,1]^X$ are N_r-contradictory if and only if

$$X_{\mu\sigma} \subset \{(y_1,y_2) \in [0,1]^2 : y_1^r + y_2^r \le 1\}$$

For each r>0 the curve $y_1^r + y_2^r = 1$ is the border that delimits the region of contradiction, and if $X_{\mu\sigma}$ takes some value $(\mu(x_0),\sigma(x_0))$ over the mentioned curve, then, they are not N_r-contradictory.

We must note that as r increases, curves $y_1^r + y_2^r = 1$ approach to the line y_1=1 (except in y_2=1) and to the line y_2=1 (except in y_1=1); more specifically, the family of functions $\{(1-y_1^r)^{1/r}\}_{r>0}$ converges punctually when $r \to \infty$, to the constant function 1 for all y_1 ∈ [0,1) and in y_1=1 converges to 0 and the family of functions $\{(1-y_2^r)^{1/r}\}_{r>0}$ converges punctually when $r \to \infty$, to the constant function 1 for all y_2∈[0,1) and in y_2=1 converges to 0; therefore, the region of non N_r-contradiction between two FS decreases when r grows (see figure 3). Moreover, when $r \to 0$, the family of functions $\{(1-y_2^r)^{1/r}\}_{r>0}$ converges for each y_2 ∈ (0,1] to the null function and for y_2=0 converges to 1; and the family of functions $\{(1-y_1^r)^{1/r}\}_{r>0}$ converges for each y_1 ∈ (0,1] to the null function and for y_1=0 converges to 1. That is, as r decreases the curves that delimit the regions of contradiction get closer to the axes y_1 and y_2, and therefore, the region of non N_r-contradiction between two FS increases.

On the other hand, if 0<r<s then, the curve $y_1^s + y_2^s = 1$ is over curve $y_1^r + y_2^r = 1$ (see figure 3 for the representation of some of them) and so, if μ, σ ∈ $[0,1]^X$ are N_r-contradictory, then they are N_s-contradictory for all s>r. In fact, if r<s it is $y_1^r > y_1^s$ for all y_1 ∈ (0,1), and therefore taking into account that $g_{\frac{1}{r}}$ is increasing and that 1/s<1/r, is $(1-y_1^r)^{1/r} < (1-y_1^s)^{1/r} < (1-y_1^s)^{1/s}$ from where we follow that coordinate y_2 of the curve corresponding to s is bigger than the one corresponding to r. Finally, we observe that the family of curves mentioned above, practically fills the unit square $[0,1]^2$ (with the exception of the border of the unit square except point (0,1) and (1,0)), That is:

$$\bigcup_{r>0}\left\{(y_1,y_2)\in[0,1]^2 : y_1^r + y_2^r = 1\right\} = (0,1)^2 \cup \left\{(1,0),(0,1)\right\}$$

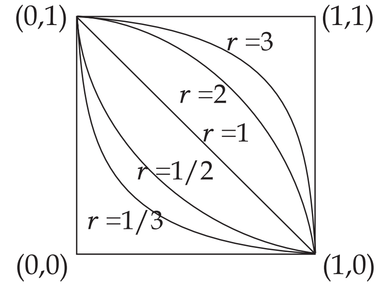


Figure 3: Curves $y_1^r + y_2^r = 1$

(d) General case of N-contradiction

If N is a strong FN, two sets μ, σ ∈ $[0,1]^X$ are N-contradictory if and only if

$$X_{\mu\sigma} \subset \left\{(y_1,y_2)\in[0,1]^2 : y_1 \le N(y_2)\right\}$$

And the border curve that delimits the region exempt of contradiction is the curve of equation $y_1=N(y_2)$. Therefore, the border curve is determined by a strong negation and it will have the following properties of a strong negation:

1) It is decreasing in both variable y_1 and y_2.
2) It goes through (1,0) and through (0,1) since N(0)=1 y N(1)=0.
3) It is symmetric with respect to the line $y_1=y_2$ since $y_1=N(y_2)$ and $y_2=N(y_1)$ are the same curve, because N(N(y))=y for all y ∈ [0,1].

Then, the regions of contradiction are limited by all strong negations in [0,1].

Degrees of N-contradiction between Two Fuzzy Sets

As we discussed in the introduction, it is relevant to weight in which degree two sets are contradictory. In fact, $\mu_{\emptyset}$ and $\mu_{\emptyset}$ (where $\mu_{\emptyset}(x)=0$ for all x ∈ X) are N-contradictory for any strong FN N (see figure 4(a)). Nevertheless, if μ and σ are N-contradictory FS such that $\mu(x_0)=N(\sigma(x_0))$ for some x_0 (see figure 4(b)), and so $X_{\mu\sigma} \cap \left\{(y_1,y_2)\in[0,1]^2 : y_1 = N(y_2)\right\} \neq \emptyset$, then small disturbances over the value $(\mu(x_0),\sigma(x_0))$ could convert μ and σ into two sets very similar to the original ones but non N-contradictory. Meanwhile, small disturbances would never change the contradictory character of the empty set with itself. Thus, it seems adequate to assign 0 as the degree of N-contradiction for whichever μ and σ such that $X_{\mu\sigma} \cap \left\{(y_1,y_2)\in[0,1]^2 : y_1 \ge N(y_2)\right\} \neq \emptyset$ and a positive value, as much higher as $X_{\mu\sigma}$ is farther away from the limit curve ($y_1=N(y_2)$), in other case (see figure 4(c)).

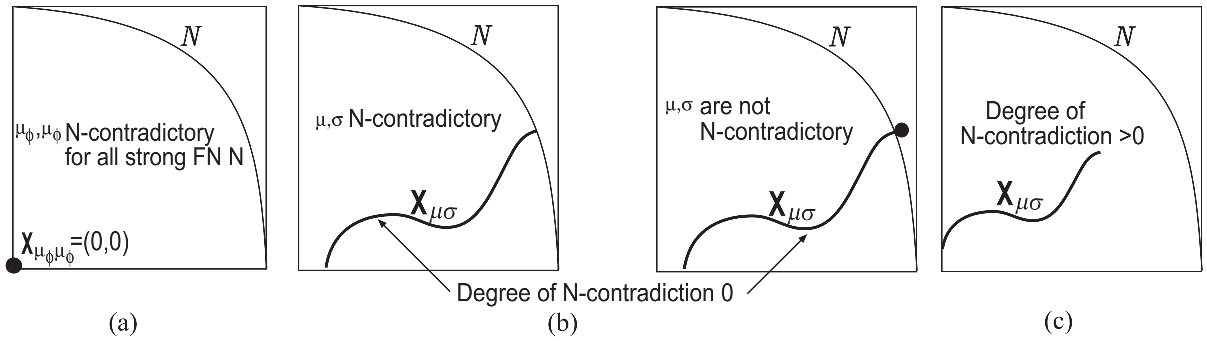

N
μ_ϕ,μ_ϕ N-contradictory for all strong FN N
$X_{\mu_\phi\mu_\phi}$=(0,0)
N
μ,σ N-contradictory
$X_{\mu\sigma}$
N
μ,σ are not N-contradictory
$X_{\mu\sigma}$
Degree of N-contradiction 0
N
Degree of N-contradiction >0
$X_{\mu\sigma}$
(a)
(b)
(c)

Figure 4: Geometrical interpretation of N-contradiction degree

Taking into account these observations, we are going to define different functions that could serve as a model to determine the different degrees of contradiction between two fuzzy sets

Definition 3.1 Given μ, σ ∈ $[0,1]^X$ and N a strong FN , we define the following contradiction measure functions:

i) $C_1^N(\mu,\sigma) = \text{Max}\left(0, \operatorname*{Inf}_{x\in X}\left(N(\sigma(x)) - \mu(x)\right)\right)$

ii) $C_2^N(\mu,\sigma) = \text{Max}\left(0, \operatorname*{Inf}_{x\in X}\left(N(\mu(x)) - \sigma(x)\right)\right)$

iii) $C_3^N(\mu,\sigma) = \text{Max}\left(0, 1 - \operatorname*{Sup}_{x\in X}\left(g(\mu(x)) + g(\sigma(x))\right)\right)$

iv) $C_4^N(\mu,\sigma) = 0$ if μ and σ are not N-contradictory, and in the other case $C_4^N(\mu,\sigma) = \dfrac{d(\text{X}_{\mu\sigma}, L_N)}{d((0,0), L_N)}$

where *d* is the Euclidean distance and $L_N = \{(y_1, y_2) \in [0,1]^2 : N(y_1) = y_2\}$ is the limit curve, and therefore, $d(\text{X}_{\mu\sigma}, L_N) = \text{Inf}\,\{d((\mu(x),\sigma(x)),\,(y_1,y_2)) : x \in \text{X}, (y_1,y_2) \in L_N\}$ and $d((0,0), L_N)\ d((0,0), L_N) = \text{Inf}\,\{d((0,0),(y_1,y_2)) : (y_1,y_2) \in L_N\}$.

Remark: The four previous functions take values in [0,1] and it is satisfied that all of them are zero or all are strictly positive. The functions C_1^N and C_2^N come motivated by the characterization of contradiction "μ and σ are N_g-contradictory if and only if μ(x)≤ N_g(σ(x)) ∀ x ∈ X ⇔ σ(x)≤ N_g(μ(x)) ∀ x ∈ X", while C_3^N is based on the characterization "μ and σ are N_g-contradictory if and only if g(μ(x))+g(σ(x))≤1 ∀x∈X". Although both characterizations are equivalent, C_1^N , C_2^N and C_3^N they do not coincide, as we will show it at a later example. On the other hand, C_4^N represents a relative distance: the Euclidean distance of the set $\text{X}_{\mu\sigma}$ to the limit curve relative to the distance of the "most contradictory" sets to the same curve. While C_1^N represents the infimum of the distances between the abscises of the values (μ(x),σ(x)) and the corresponding of the limit curve (see figure 5), C_2^N represents the infimum of the distances between the ordinates of the values (μ(x),σ(x)) and the corresponding of the limit curve (see figure 5). As far as C_3^N is concerned, some geometrical interpretations can be found in some particular cases.

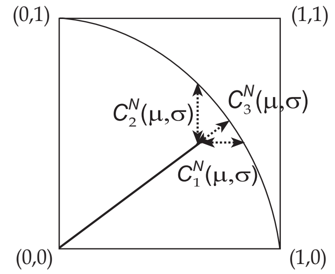


Figure 5: Geometrical interpretation of different contradiction degrees

Proposition 3.2 Let N_{id} be the standard FN; for all μ, σ ∈ $[0,1]^X$ the degrees of contradiction between μ and σ by means of the formula in definition 3.1 satisfy that $C_1^{N_{Id}}(\mu,\sigma) = C_2^{N_{Id}}(\mu,\sigma) = C_3^{N_{Id}}(\mu,\sigma) = C_4^{N_{Id}}(\mu,\sigma)$ (fig.6).

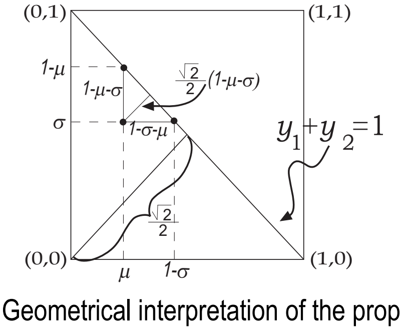


Figure 6: Geometrical interpretation of the proposition 3.2

However, generally, the four measures are different as the following examples show.

Example 3.3 Given $X_{\mu\sigma}$= {(0.25,0.75), (0.65,0.65), (0.75,0.5), (0.79,0.3)} and the strong negation $N_3(y) = \left(1 - y^3\right)^{1/3}$, with g(y)=y^3. Then (see figure 7):

$$C_1^{N_3}(\mu,\sigma) = \operatorname*{Inf}_{x \in X}\left(\left(1 - \sigma(x)^3\right)^{1/3} - \mu(x)\right) = 0.2009$$ reaching the infimum at point (0.79, 0.3)

$$C_2^{N_3}(\mu,\sigma) = \operatorname*{Inf}_{x \in X}\left(\left(1 - \mu(x)^3\right)^{1/3} - \sigma(x)\right) = 0.2448$$ reaching the infimum at point (0.25, 0.75)

$$C_3^{N_3}(\mu,\sigma) = 1 - \operatorname*{Sup}_{x \in X}\left(\mu(x)^3 + \sigma(x)^3\right) = 0.4507$$ reaching the supremum at point (0.65, 0.65)

Since μ and σ are N$_3$-contradictory and since $L_N = \left\{(y_1, y_2) \in [0,1]^2 : {y_1}^3 + {y_2}^3 = 1\right\}$, then

$$C_4^{N_3}(\mu,\sigma) = \frac{d(X_{\mu\sigma}, L_N)}{d((0,0), L_N)} = d\left((0.79,0.06), L_N\right) = 0.1969$$

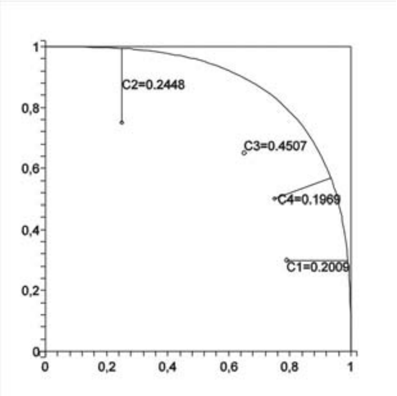


Figure 7: Geometrical interpretation of the example 3.3

Proposition 3.4 Let N$_g$ be the strong FN with g(y)=y^2, i.e. $N_g(y) = \sqrt{1 - y^2}$, then for all μ, σ ∈ [0,1]^X the degrees of contradiction $C_3^{N_g}$ and $C_4^{N_g}$ between μ and σ verify that $C_3^{N_g}(\mu,\sigma) = 1 - \left(1 - C_4^{N_g}(\mu,\sigma)\right)^2$ (see figure 8).

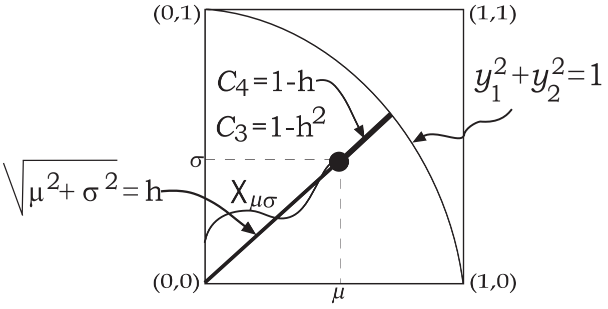


Figure 8: Geometrical interpretation of the proposition 3.4

Let us observe that, in general, the relation of the proposition 3.3 is not satisfied as the example 3.3 shows:

$$C_3^{N_3}(\mu,\sigma) = 0.4507 \neq 1 - \left(1 - C_4^{N_3}(\mu,\sigma)\right)^2 = 1 - (1 - 0.1969)^2 = 0.3550$$

The following properties of the above measure of N-contradiction functions between two fuzzy sets can be proved.

Proposition 3.5 For each i=1,2,3,4 function C_i^N : [0,1]^X x [0,1]^X → [0,1] defined for every two μ, σ ∈[0,1]^X as definition 3.1 verifies:

i) $C_i^N(\mu_\emptyset, \mu_\emptyset) = 1$.

ii) $C_i^N(\mu,\sigma) = 0$ if μ or σ normal.

iii) Symmetry: $C_i^N(\mu,\sigma) = C_i^N(\sigma,\mu)$ for i=3,4. For i=1,2 is verified that $C_1^N(\mu,\sigma) = C_2^N(\sigma,\mu)$.

iv) Given $\{\mu_\alpha\}_{\alpha\in I}\subset[0,1]^X$, it holds that: $\underset{\alpha\in I}{\text{Inf}}\, C_i^N(\mu_\alpha,\sigma) = C_i^N\left(\underset{\alpha\in I}{\text{Sup}}\,\mu_\alpha,\sigma\right)$. As a particular case of (iv) it is verified that given μ_1, μ_2,$\sigma\in[0,1]^X$ if $\mu_1 \le \mu_2$, then $C_i^N(\mu_1,\sigma) \ge C_i^N(\mu_2,\sigma)$ (Anti-Monotonicity).

Property (ii) is stronger than the second axiom given in [3] ($C(\mu,\mu)=0$ for all normal $\mu \in [0,1]^X$) to define measures of contradiction. Moreover, C_i^N for each i=1,2,3,4, is a positive or strict measure of contradiction as defined in [3] since $C_i^N(\mu,\sigma)=0$ provided that $\underset{x\in X}{\text{Sup}}\big(g(\mu(x)) + g(\sigma(x))\big) \ge 1$.

Example 3.6 Given *P, Q* $\in\mathcal{F}([0,1])$ with membership functions μ, σ such that $\mu(x)= -2x+1$ if $x\le 1/2$ and 0, if $x>1/2$ and $\sigma(x)= x$, if $x\le 1/2$ and 1/2, if $x>1/2$. As μ is normal, for all strong negation N the degree of contradiction is zero, $C_i^N(\mu,\sigma)=0$ with i=1,2,3,4. However, there are strong negations for which sets P, Q are contradictory. For instance, for all negations N_g such that $g(y)=y^p$ with $p\ge 1$.

Measuring Contradiction between Two Fuzzy Sets

In this section, we will deal with the case of contradiction without depending on a prefixed negation. The previous section establishes the contradiction between two fuzzy sets related to a chosen strong negation. We now address contradiction more generally, without depending on any specific FN. In [5] and [6] two FS *P, Q* $\in\mathcal{F}(X)$ with membership functions μ, σ were defined contradictory if they were N-contradictory regarding some strong FN N. The following result was proved in [2].

Proposition 4.1 ([2]) If *P, Q* $\in\mathcal{F}(X)$ with membership functions μ, σ are contradictory, then: for all $\{x_n\}_{n\in N}\subset X$, if $\lim_{n\to\infty}\{\mu(x_n)\}=1$, then $\lim_{n\to\infty}\{\sigma(x_n)\}=0$, and if $\lim_{n\to\infty}\{\sigma(x_n)\}=1$, then $\lim_{n\to\infty}\{\mu(x_n)\}=0$. In particular, if $\mu(x)=1$, for some $x \in X$, then $\sigma(x)=0$ and vice-versa.

With the intention of measuring how contradictory two FS are, we will define some functions motivated in the previous section, being of interest, for one of them, to consider the following corollary also given in [2].

Corollary 4.2 ([2]) If μ, $\sigma \in [0,1]^X$ are contradictory, then $\underset{x\in X}{\text{Sup}}\big(\mu(x) + \sigma(x)\big) < 2$.

Definition 4.3 Given μ, $\sigma \in [0,1]^X$, we define the following contradiction measure functions:

i) $C_1(\mu,\sigma)=0$ if there exists $\{x_n\}_{n\in N} \subset X$ such that $\lim_{n\to\infty}\{\mu(x_n)\}=1$ or $\lim_{n\to\infty}\{\sigma(x_n)\}=1$, and, in other case $C_1(\mu,\sigma) = \text{Min}\Big(\underset{x\in X}{\text{Inf}}(1-\mu(x)), \underset{x\in X}{\text{Inf}}(1-\sigma(x))\Big)$.

ii) $C_2(\mu,\sigma)=0$ if there exists $\{x_n\}_{n\in N} \subset X$ such that $\lim_{n\to\infty}\{\mu(x_n)\}=1$ or $\lim_{n\to\infty}\{\sigma(x_n)\}=1$, and, in other case $C_2(\mu,\sigma) = 1 - \dfrac{\underset{x\in X}{\text{Sup}}\big(\mu(x)+\sigma(x)\big)}{2}$.

Remark: It is evident that the function C_1 measures the minimum between distance (Euclidean) of $X_{\mu\sigma}$ to the line $y_1=1$ (that we will note L_1) and the distance of $X_{\mu\sigma}$ to the line $y_2=1$ (that we will note L_2): (see figure 9(a))

$$C_1(\mu,\sigma) = \text{Min}\big(d(X_{\mu\sigma},L_1), d(X_{\mu\sigma},L_2)\big) = \frac{\text{Min}\big(d(X_{\mu\sigma},L_1), d(X_{\mu\sigma},L_2)\big)}{d((0,0), L_1 ó L_2)}.$$

On the other hand, $C_2(\mu,\sigma) = \dfrac{d_1(X_{\mu\sigma},(1,1))}{2} = \dfrac{d_1(X_{\mu\sigma},(1,1))}{d_1((0,0),(1,1))}$ that is, the function C_2 measures the reticular distance between $X_{\mu\sigma}$ and (1,1), relative to the reticular distance from (0,0) to (1,1) (let us remind that $d_1\big((y_1,y_2),(z_1,z_2)\big) = |y_1 - z_1| + |y_2 - z_2|$) (see figure 9(b)). These geometrical interpretations of the measures C_1 and C_2 suggest another way of measuring the contradiction degree: $C_3(\mu,\sigma)=0$ if there exists

$\{x_n\}_{n\in N}\subset$ X such that $\lim\limits_{n\to\infty}\{\mu(x_n)\}=1$ or $\lim\limits_{n\to\infty}\{\sigma(x_n)\}=1$, and, in other case $C_3(\mu,\sigma)=\dfrac{d(X_{\mu\sigma},(1,1))}{d((0,0),(1,1))}$ (see figure 9(c)).

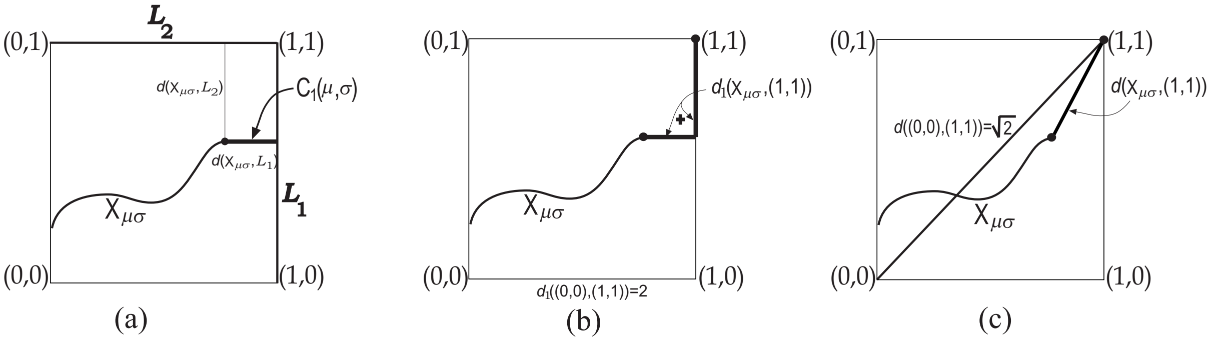


Figure 9: Geometrical interpretation of the measures C_1, C_2 and C_3

In the same way that happened with measures of N-contradiction between two fuzzy sets, the following result can be demonstrated.

Proposition 4.4 For each i=1,2,3 function C_i: $[0,1]^X \times [0,1]^X \to [0,1]$ defined for each pair μ, $\sigma\in[0,1]^X$ as the above definition verifies:

i) $C_i(\mu_\varnothing,\mu_\varnothing)=1$.

ii) $C_i(\mu,\sigma)=0$ if μ or σ normal.

iii) Symmetry: $C_i(\mu,\sigma)=C_i(\sigma,\mu)$.

iv) Anti-Monotonicity: given μ_1, μ_2, $\sigma\in[0,1]^X$ if $\mu_1\le\mu_2$, then $C_i(\mu_1,\sigma)\ge C_i(\mu_2,\sigma)$. Besides, for the case i=1 axiom of the infimum given in [3] is also verified. That is, given $\{\mu_\alpha\}_{\alpha\in I}\subset[0,1]^X$, it holds that: $\operatorname*{Inf}\limits_{\alpha\in I} C_i(\mu_\alpha,\sigma)=C_i\left(\operatorname*{Sup}\limits_{\alpha\in I}\mu_\alpha,\sigma\right)$.

Bibliography

[1] E. Castiñeira, S. Cubillo and S. Bellido. Degrees of Contradiction in Fuzzy Sets Theory. Proceedings IPMU'02, 171-176. Annecy (France), 2002.

[2] E. Castiñeira, S. Cubillo. and S. Bellido. Contradicción entre dos conjuntos. Actas ESTYLF'02, 379-383. León (Spain), 2002, (in Spanish).

[3] S. Cubillo and E. Castiñeira. Measuring contradiction in fuzzy logic. International Journal of General Systems, Vol. 34, Nº1, 39-59, 2005.

[4] E. Trillas. Sobre funciones de negación en la teoría de conjuntos difusos. Stochastica III/1, 47-60, 1979 (in Spanish). Reprinted (English version) (1998) in Avances of Fuzzy Logic. Eds. S. Barro et altr, 31-43.

[5] E. Trillas, C. Alsina and J. Jacas. On Contradiction in Fuzzy Logic. Soft Computing, 3(4), 197-199, 1999.

[6] E. Trillas and S. Cubillo. On Non-Contradictory Input/Output Couples in Zadeh's CRI. Proceedings NAFIPS, 28-32. New York, 1999.

[7] L. A. Zadeh. Fuzzy Sets. Inf. Control, volume 20, pages 301-312, 1965.

Authors' Information

Carmen Torres – Dept. Applied Mathematic. Computer Science School of University Politécnica of Madrid. Campus Montegancedo. 28660 Boadilla del Monte (Madrid). Spain; e-mail: ctorres@fi.upm.es

Elena Castiñeira – Dept. Applied Mathematic. Computer Science School of University Politécnica of Madrid. Campus Montegancedo. 28660 Boadilla del Monte (Madrid). Spain; e-mail: ecastineira@fi.upm.es

Susana Cubillo – Dept. Applied Mathematic. Computer Science School of University Politécnica of Madrid. Campus Montegancedo. 28660 Boadilla del Monte (Madrid). Spain; e-mail: scubillo@fi.upm.es

Victoria Zarzosa – Dept. Applied Mathematic. Computer Science School of University Politécnica of Madrid. Campus Montegancedo. 28660 Boadilla del Monte (Madrid). Spain; e-mail: vzarzosa@fi.upm.es

A WORKBENCH FOR DOCUMENT PROCESSING

Karola Witschurke

Abstract: During the MEMORIAL project time an international consortium has developed a software solution called DDW (Digital Document Workbench). It provides a set of tools to support the process of digitisation of documents from the scanning up to the retrievable presentation of the content. The attention is focused to machine typed archival documents. One of the important features is the evaluation of quality in each step of the process. The workbench consists of automatic parts as well as of parts which request human activity. The measurable improvement of 20% shows the approach is successful.

Keywords: Document Management, Digital Document Workbench, Image Processing, OCR, Machine Typed Document.

ACM Classification Keywords: I.7.5 Document and Text Processing: Document Capture

Introduction

A strategic goal of the international consortium undertaking this project was to support the creation of virtual archives based on documents which exist in libraries, archives, museums, memorials and public record offices, in order to allow computer aided information retrieval.

Machine type written documents have proved as especially hard to process.

Such documents may constitute less or more complex printed forms mixing printed and typed text, graphics, as well as hand written annotations, signatures, rubber stamps and photographs. Moreover, the colour of typed characters may vary; characters may be overstricken, shifted up or down, or due to torn out or dried ribbon only partially typed. A special challenge are documents which represent a carbon copy instead of an original one, Finally, due to physical conditions a document may contain stained or damaged parts.

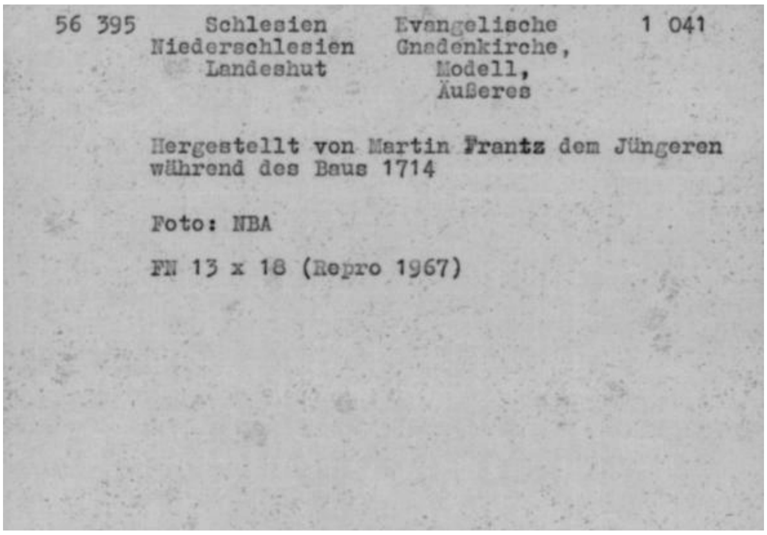
1. Approach to the Project

OCR systems may be used to extract the textual information from a document image.

OCR is used to working on binary images and assigns groups of black pixels to patterns of characters. The results of state-of-the-art OCR systems are satisfying if fresh printed office documents are processed. Historical documents in general look bad for different reasons.

The problem is to get good looking binary images from bad looking coloured original ones by filtering noise (stains, wrinklings, torns) out of the image and improving the shapes of characters. Due to the unsteady quality of a page it is not satisfactory to use the same threshold of binarisation (a parameter between 1 and 256 used for binarisation of a gray scale image and causes the assignment of pixel to be “white” or “black”) for the whole page. For those reason in the MEMORIAL-project a semantic driven approach was preferred. A special editor supports the user to draw regions of interest and mark other regions with damages or illegible parts. After that a partial image improvement and background clearing using the colour information is possible. Thus, the binarisation with the best possible threshold is processed adaptively for the different regions down to single characters. The subsequent figures are illustrating the advantage of this approach.

The often automatically chosen binarsation threshold of 128 in this case provides a nearly blank image caused by the weakness of the typed characters. Figure 2 compares three manually chosen thresholds with the adaptively applied threshold of DDW. The marked region in figure 3 shows the advance of DDW vs. the best manual binarisation. In this case, the OCR is getting the best possible input.



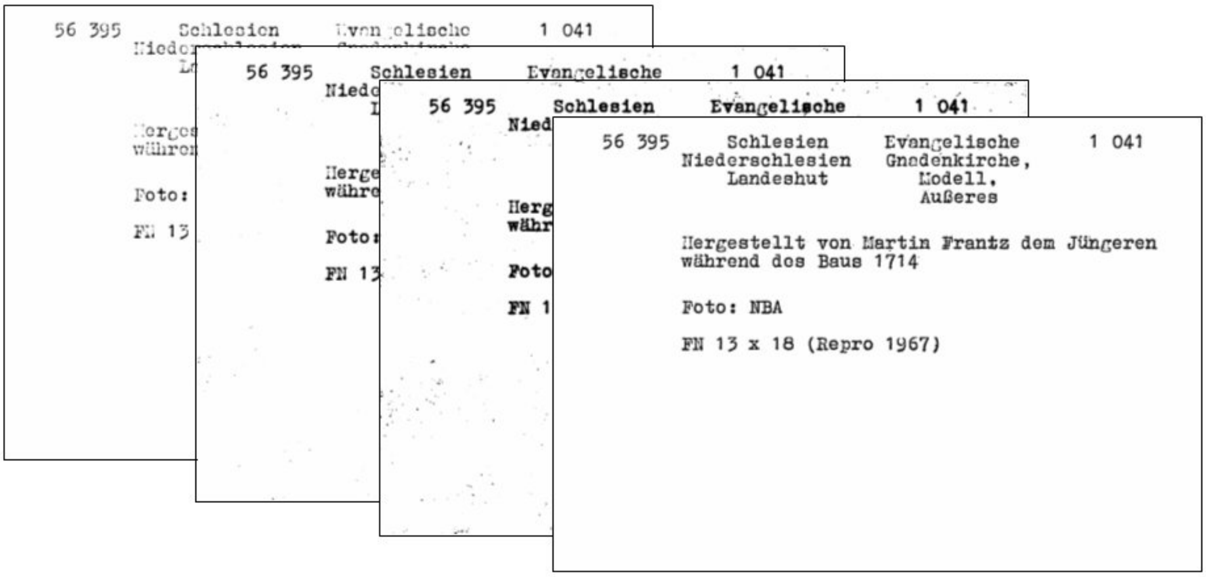
56 395 Schlesien Evangelische 1 041
Niederschlesien Gnadenkirche,
Landeshut Modell,
Äußeres

Hergestellt von Martin Frantz dem Jüngeren
während des Baus 1714

Foto: NBA

FN 13 x 18 (Repro 1967)

Figure 1 – inventory card from Herder-Institute Marburg (Germany)



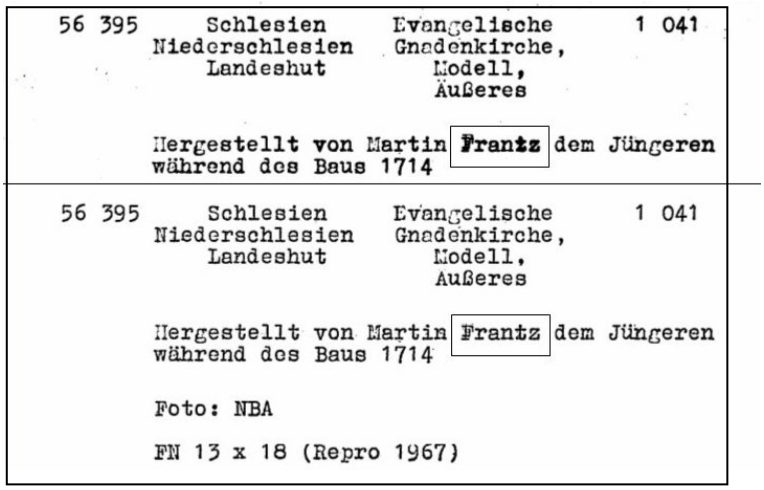
56 395 Schlesien Evangelische 1 041
Niederschlesien Gnadenkirche,
Landeshut Modell,
Außeres

Hergestellt von Martin Frantz dem Jüngeren
während des Baus 1714

Foto: NBA

FN 13 x 18 (Repro 1967)

Figure 2 – comparisation of threshold (from left) 180, 200, 210 with adaptive threshold used by DDW (right)



56 395 Schlesien Evangelische 1 041
Niederschlesien Gnadenkirche,
Landeshut Modell,
Außeres

Hergestellt von Martin Frantz dem Jüngeren
während des Baus 1714

Foto: NBA

FN 13 x 18 (Repro 1967)

Figure 3 – best manual binarisation (above) versus adaptive thresholding

2. The DDLC Model

The transformation of a historical paper document into its electronic parallel is a complex multi-phased engineering process which may be interpreted as a document life cycle. In this effect, the project consortium has developed a *Digital Document Workbench (DDW)* toolkit supporting a *Digital Document Life-Cycle Development (DDLC)* model.

The first phase of the DDLC model is *digitisation*, which provides a raw digital image of a paper original. Depending on the constitution of the documents this process may be done manually or semiautomatically. The quality in this phase immediately influences the quality of the following phases. Furthermore the naming of the image files happens in this phase (mostly by the scanner) – each generated file must have a unique name to avoid processing duplicates or overwriting files during their processing later on. A document *Repository Management Tool (RMT)* of the DDW toolkit has been developed to help the archivist in the namespace management and to store images in a database.

The second phase of DDLC is *qualification.* An expert user should attentively classify documents by building groups of similar in structure and meaning documents. Documents within the same semantic class can be processed together throughout the rest of the cycle. The output of this phase is a XML structure called *document template*, which contains the formal description of a semantic class.

The third phase of DDLC is *segmentation*, where the identification of major components (regions) of a document page image happens. The document template is used to control the segmentation phase, where the raw document image is cleaned and improved by the *Image Processing Tool (IPT)* which transforms original (coloured) TIFF files into binarised clean images. The output of this phase is a *document content* XML file, which represents an interface for the next steps.

The fourth phase of DDLC is *extraction*, a key phase of the DDLC. Here the clean document image is processed by OCR; i.e. the textual information of an image is transformed into computer text. In this phase, the *document content* XML file is filled with the results of OCR.

The following *acceptance* phase allows the user to decide whether the recognised text is suitable to be introduced into the target database (digital archive). Corrections should be done to deliver the essential quality for the subsequent exploitation phase. This effort is supported by the content editor *Generator of Electronic Documents (GED)*. The multivalent browser *Viewer of Electronic Documents (VED)* futhermore enables addition of notifications. This might be required to improve the quality of the results of queries to the target database.

The graphical representation of the DDLC model with its interfaces is shown in figure 4.

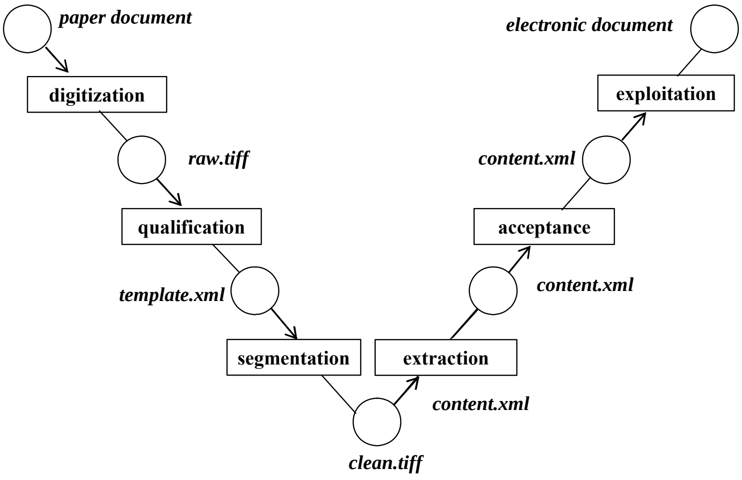


Figure 4 - DDLC model

The left part of the DDLC model represents analysis of information aided by the user, whose domain knowledge is gradually being transformed into a control structure of processes for engineering the final product, represented by the right part. Verification of partial products of respective phases of the cycle plays a key role in assuring the quality of the final product. DDW tools enable a great deal of flexibility in extracting content of scanned paper documents into electronic documents. A sample representative subset of documents constituting a class can be processed in a *semiautomatic mode* with DDW tools to find the best settings of parameters for each DDLC phase, and next the remaining documents of the same class can be processed automatically in a batch mode, with the same settings. This approach introduced by the consortium enables fine-tuning of a document content extraction processes and quality management throughout the entire cycle.

3. Quality Management

In order to assure the best possible quality in each step of the DDLC model a quality management has been established. Therefore, human expertise is requested. The QED Tool of the DDW supports the fine-tuning of the process displaying the parameters and metrics on the one hand and setting up weights on the other hand. The interface for quality data exchange is the *qed.xml* file which is stored in the working repository (see figure 5 and table 1).

When the quality of the paper document is *Q(PD)* and quality of the electronic counterpart is *Q(ED)* then three relations are possible.

Q(PD) > *Q(ED)*, the final product quality has deteriorated during processing along DDLC phases;

Q(PD) $\cong$ *Q(ED)*, the final document quality has not significantly changed compared to the original;

Q(PD) < *Q(ED)*, the final document quality has been improved during processing.

The first case indicates incorrect parameter settings. The achieved electronic document is unacceptable. The second case indicates correct but not optimal parameter settings. The third case is the desirable one - indicating an increasing quality during processing along DDLC. It is possible only when an expert user has been able to successfully contribute to the document engineering processes.

Document quality assessment in any DDLC phase uses a specially developed Visual GQM (VGQM) method [9]. The VGQM method distinguishes between parameters and metrics. Parameters characterise processes of each phase, and their values may be used to control the DDW components. The values of metrics, specific to each phase, are measured to characterise the particular input and output data. The value of Q is calculated based on a quality tree and normalised to a five-grade scale, from very low, through low and medium, up to high, and very high quality. While all optimal settings for each phase are established by the quality expert, the processing of the remaining documents of the class can be performed automatically in a batch by an archivist. Any document that cannot pass the quality threshold set up by an expert may now be rejected. In dependence on the acceptance, a phase may rerun with tuned parameters. A thorough selection of acceptance criteria for each class implies that either document processing progresses to the next phase, or is of such a poor quality that it must be processed manually (retyped).

4. Digital Document Workbench

DDW is a set of tools which aids the user in transferring typed content of paper documents into a digital archive. Any realistic use of DDW is possible, if the following assumptions are valid:

- all paper documents are type written;
- originals may be yellowed, dirty and otherwise bad looking;
- many documents which match the same layout may be found in the paper archive;
- a target digital archive should contain machine readable text of the documents.

The "backbone" of the DDW is a MS-SQL database (of-the-shelf product) called a Working Repository (**WR**). It contains information on the entire lifecycle of each document. Component tools post their results into the working repository, this way information exchange is guaranteed between all tools. Figure 5 gives summary of the several tools belonging to the DDW and the ways of information exchange.

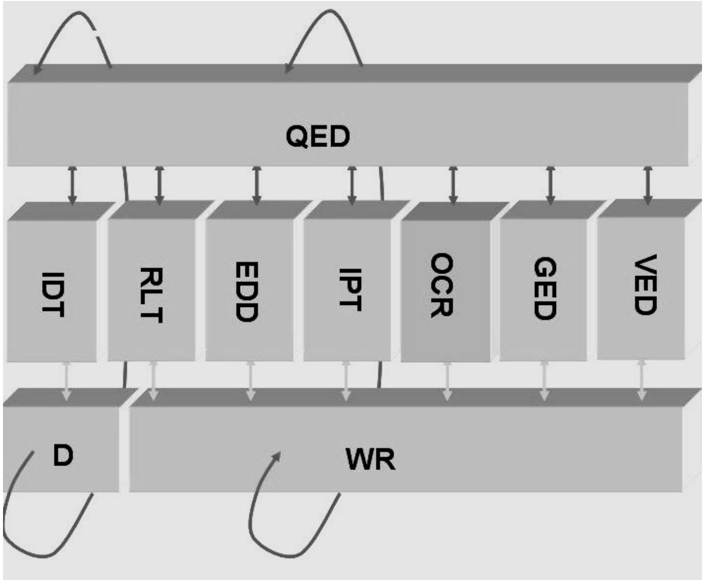


Figure 5 – Architecture of the DDW Toolset

Acronym	*Full name*	*Functionality*
IDT	**I**n**D**exing **T**ool	Supportes name management, storage management and meta data description
RLT	**R**epository **L**oading **T**ool	Stores metadata and links to the image files as well as automatically generated jpeg-files and thumbnails in the working repository.
EDD	**E**lectronic **D**ocument class template e**D**itor	Creates and edits a template.xml file to describe a document class, allows to add (similar) documents to the class
IPT	**I**mage **P**rocessing **T**ool	Performs background cleaning and character improvement, creates a content.xml file
OCR	**O**ptical **C**haracter **R**ecognition tool	Separates the information relevant for OCR from the content.xml, runs the OCR (of-the-shelf product) and returns the results of OCR to the content.xml
GED	**G**enerator of **E**lectronic **D**ocuments	Enables editing badly recognised text or (in a extreme case) retyping a document
VED	**V**iewer of **E**lectronic **D**ocuments	Allows the archivist to browse layers of the electronic document with lenses.
QED	**D**ocument **Q**uality **E**valuation Tool	Supports measurement and evaluation of the quality of each respective DDLC phase. If a quality level is not satisfactory, the whole process can be repeated with changed settings.
WR	**W**orking **R**epository	Is an internal DDW database for storing project data

Table 1 – DDW components

The DDW toolkit can be used in two possible configurations:

- stand alone, without any connection to the working repository, with all relevant files stored directly in a common file system. This can be useful when just a few documents in manual mode are processed. In this case, the user has to control the file system.
- connected to the working repository (DDW database), providing a better control on intermediary document forms in between DDLC phases, in particular when operating DDW in a batch mode.

5. Final Remarks

One of the key advances of DDW is the semantic driven image processing. The success of the approach to handle colour images of documents by developed image processing tools (IPT) is shown in the following impressive graphic. It maps the confidence rate of 50 documents (register cards of Herder - Institute Marburg) depending on the threshold of binarisation. In general, OCR systems choose t =128 as threshold automatically. It can also be chosen interactively (by testing the documents and looking for the best results, here: t = 189). In both cases, the chosen threshold then holds for all documents as a whole. DDW determines the threshold value individually for each character. Figure 6 shows the enhancement of quality processing 50 different inventory cards as displayed in figure 1 with fixed thresholds vs. with an adaptively determined threshold (upper curve, resp. right column). The obtained average value of ca. 80 is not yet satisfying for the majority of archivists, but advanced methods will increase the results in further projects.

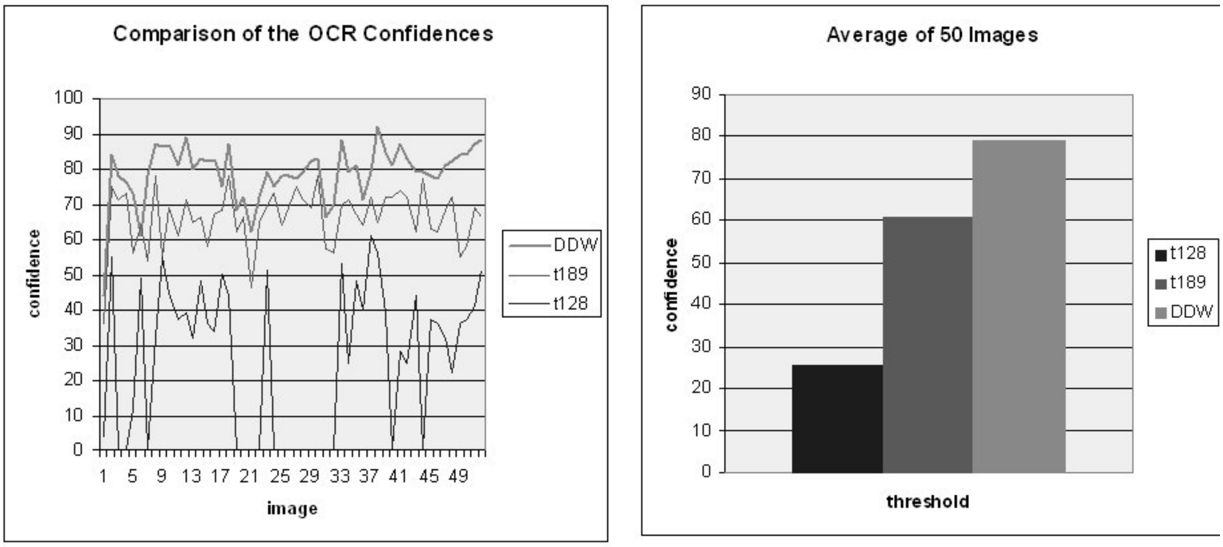


Figure 6 – Influence of binarisation parameters on the OCR results

During two workshops archivists could act as model users and test DDW with a selected set of documents. A typical situation observed during the test phase is the quality improvement combined with a significant reduction of time and effort in editing a document during the acceptance phase, compared to manual reproduction of a document from scratch.

Acknowledgements

The MEMORIAL project was funded by the European Commission within the Information Society Technologies (IST) Program from 2002-02-01 to 2004-10-31. The consortium implementing the goal statements was constituted by GfaI (Society for the Promotion of Applied Computer Science)– Germany, Moreshet Holocaust Study and Research Center– Israel, National Museum Stutthof in Sztutowo - Poland, Technical University of Gdańsk – Poland, University of Liverpool – Great Britain.

Bibliography

[1] Antonacopoulos, A., Karatzas, D., Krawczyk, H., Wiszniewski, B.; The Lifecycle of a Digital Historical Document: Structure and Content.; ACM symposium on Document Engineering, Milwaukee, USA, October 28-30, 2004

[2] A. Antonacopoulos, D. Karatzas; A Complete Approach to the conversion of typewritten Historical Documents for Digital Archives; Sixth IAPR International Workshop on Document Analysis Systems, Florence, Italy, 8-10 September 2004

[3] Szwoch M., Szwoch W.; Preprocessing and Segmentation of Bad Quality Machine Typed Paper Documents; Sixth IAPR International Workshop on Document Analysis Systems, Florence, Italy, 8-10 September 2004

[4] Apostolos Antonacopoulos; Document Image Analysis for World War II Personal Records; International Workshop on Document Image Analysis for Libraries, Palo Alto, Jan 2004

[5] Wolfgang Schade, Karola Witschurke, Cornelia Rataj; Improved character recognition of typed documents from middling and lower quality based on application depending tools Processes, results, comparison; Electronic Imaging Events in the Visual Arts - EVA 2003, Berlin 2003

[6] Alexander Geschke, Eva Fischer; Memorial Project - A complex approach to digitisation of personal records; Electronic Imaging Events in the Visual Arts - EVA 2003, Berlin 2003

[7] Henryk Krawczyk, Bogdan Wiszniewski; Definition gleichartiger Dokumententypen zur Verbesserung der Erkennbarkeit und ihre XML-Beschreibung; Electronic Imaging Events in the Visual Arts - EVA 2003, Berlin 2003

[8] Henryk Krawczyk, Bogdan Wiszniewski; Digital Document Life Cycle Development; International Symposium on Information and Communication Technologies, ISICT 2003, Dublin, Ireland

[9] Henryk Krawczyk, Bogdan Wiszniewski; Visual GQM approach to quality-driven development of electronic documents; Second International Workshop on Web Document Analysis, WDA2003, Edinburgh, UK

[10] Bogdan Wiszniewski; Projekt IST-2001-33441-MEMORIAL: Zestaw narzędziowy do tworzenia dokumentów cyfrowych z zapisów osobowych; I Krajowa Konferencja Technologii Informacyjnych 2003 TUG

[11] Alexander Geschke; MEMORIAL Project Overview; Proc. EVA Harvard, Symposium about Collaboration of Europe, Israel and USA, Harvard Library 1-2.10.2003

[12] Jacek Lebiedź, Arkadiusz Podgórski, Mariusz Szwoch; Quality Evaluation Of Computer Aided Information Retrieval From Machine Typed Paper Documents; Third conference on Computer Recognition Systems KOSYR'2003

[13] Witold Malina, Bogdan Wiszniewski; Multimedialne biblioteki cyfrowe; Sesja 50-lecia WETI-PG

[14] Dr. Alexander Geschke, Dr. Wolfgang Schade; The EU Project Memorial - Digitisation, Access, Preservation; Electronic Imaging Events in the Visual Arts - EVA 2002, Berlin 2002

[15] S. Rogerson, B. Wiszniewski; Legislation and regulation: emphasis on European approach to Data Protection, Human Rights, Freedom of Information, Intellectual Property, and Computer Abuse; PROFESSIONALISM IN SOFTWARE ENGINEERING PSE'03

Author's Information

Karola Witschurke - GFaI, Rudower Chaussee 30, 12489 Berlin, Germany; e-mail: witschurke@gfai.de

EXPERIMENTS IN DETECTION AND CORRECTION OF RUSSIAN MALAPROPISMS BY MEANS OF THE WEB

Elena Bolshakova, Igor Bolshakov, Alexey Kotlyarov

Abstract: Malapropism is a semantic error that is hardly detectable because it usually retains syntactical links between words in the sentence but replaces one content word by a similar word with quite different meaning. A method of automatic detection of malapropisms is described, based on Web statistics and a specially defined Semantic Compatibility Index (SCI). For correction of the detected errors, special dictionaries and heuristic rules are proposed, which retains only a few highly SCI-ranked correction candidates for the user's selection. Experiments on Web-assisted detection and correction of Russian malapropisms are reported, demonstrating efficacy of the described method.

Keywords: semantic error, malapropism, error correction, Web-assisted error detection, paronymy dictionaries, correction candidates, Semantic Compatibility Index.

ACM Classification Keywords: I.2.7 [Artificial Intelligence]: Natural language processing – Text analysis

Introduction

Modern computer text editors and spellers readily detect spelling errors and some syntactic errors, primarily, mistakes in word agreement. Step by step, editing facilities of computers are being extended, in particular, by taking into account specificity of the particular text style and genre [4]. The topical problem is now semantic mistakes, which are hardly detectable because they violate neither orthography nor grammar of the text.

Malapropism is a particular type of semantic mistakes, which replace one content word by another similar word. The latter has the same morpho-syntactic form but different meaning, which is inappropriate in the given context, e.g., *animal word* or *massy migration* given instead of *animal world* and *massive migration*. To correct such mistakes, computer procedures are required that reveal erroneous words and supply the user (human editor) with selected candidates for their correction. However, only few papers (cf. [3, 5]) are devoted to the problem of malapropism detection and correction.

A method for malapropism detection proposed in [5] relies on recognition in the text of words (mainly nouns) distant from all contextual ones in terms of WordNet semantic relations (of synonymy, hyponymy, hyperonymy, etc.). Syntactic relations between words are ignored, and words from different sentences or even paragraphs are analyzed.

In the paper [3] malapropism detection is based on syntactico-semantic relations between content words, thereby much smaller context – only one sentence – is needed for error detection. Specifically, sentences are considered consisting of syntactically related and semantically compatible combinations of content word, the so-called *collocations*. It is presumed that malapropisms destroy collocations they are in: they violate semantic compatibility of word combinations while retaining their syntactic correctness.

In order to detect errors in a sentence, all pairs of syntactically linked content words in it are verified as collocations: their semantic compatibility is tested. Words of four principal parts of speech (POS) – nouns, verbs, adjective, and adverb – are considered as collocation components. To test whether a word pair is a collocation, three types of linguistic resources are proposed: a precompiled collocation database like CrossLexica [1], a text corpus, or a Web search engine like Google or Yandex.

This paper develops the latter method on the basis of experiments with Yandex as a resource for collocation testing. The Web is widely considered now as a huge, but noisy linguistic resource [6]. For the Web, it proved necessary to revise heuristic rules for malapropism detection and correction.

Following [3] we consider only malapropisms that destroy collocations. A malapropism is detected if a pair of syntactically linked content words in a sentence exhibits the value of a specially defined *Semantic Compatibility Index* (SCI) lower than a predetermined threshold. Below we call malapropism the whole pair detected as erroneous.

For malapropism correction, in contrast with the blind search of editing variants used in [3, 5], we propose to use beforehand compiled dictionaries of paronyms, i.e. words differing in some letters or in some morphs. The dictionaries provide all possible candidates for correction of a malapropos word, and the candidates are then tested in order to select several highly SCI-ranked correction candidates for ultimate decision by the user.

The proposed method was examined on two sets of Russian malapropisms. The first set of a hundred of samples was used to adjust heuristic threshold values, whereas the second justified these values. Since collocation components (hereafter *collocatives*) may be adjacent or separated by other words in a sentence, in the experiment we took into account the most probable distances between collocatives, which were previously determined through Yandex statistics.

Dictionaries of Paronyms

For correction of malapropos words, quick search of similar words are required. Words similar in letters, sounds or morphs, are usually called paronyms. In any language, only a limited portion of words has paronyms, and paronymy groups are on an average rather small. Hence, it is reasonable to gather paronyms before their use.

For our purposes, we consider only ***literal*** (e.g., Eng. *pace* Vs. *pact*, or Rus. *краска* Vs. *каска*) and ***morphemic*** paronyms (e.g., Eng. *sensible* Vs. *sensitive*, or Rus. *человечный* Vs. *человеческий*). Russian paronyms of these two types were compiled in corresponding dictionaries, which were preliminary described in [2].

The dictionary of Russian literal paronyms consists of word groups. Each group includes an entry word and its one-letter paronyms. Such paronyms are obtained through applying to the entry word of an elementary editing operation: insertion of a letter in any position, omission of a letter, replacing of a letter by another one, and permutation of two adjacent letters. For example, Russian word *белка* has one-letter paronymy group {*булка*, *елка*, *телка*, *челка*, *щелка*}.

The paronymy groups include words of the same part of speech (POS). Moreover, nouns of singular number and nouns of plural number, as well as nouns for different genders of singular have separated groups. Similar division is done for personal and other forms of verbs. Such measure is necessary for retaining syntactic links between words in the sentence while correcting an erroneous word. For this purpose, we extract malapropos word from the text (e.g., Rus. *белкой*), reconstruct its dictionary morphological form (*белка*), take from the dictionary corresponding paronym (e.g., *булка*), change its morphological form (taking it the same as for sourse malapropos word), and, finally, replace the erroneous word by the resulted word (*булкой*).

An entry of dictionary of Russian morphemic paronyms presents a group of words of the same POS that have the same root morph but differ in auxiliary morphs (prefixes or suffixes), e.g. Rus. {*бегающий*, *беглый*, *беговой*, *бегущий*}.

By now, the developed dictionary of literal paronyms comprises 17,4 thousands of paronymy groups with the mean size 2,65, while the dictionary of morphemic paronyms contains 1310 groups with the mean size 7,1.

Method of Malapropism Detection and Correction

To facilitate understanding of key ideas of the method we should first clarify the notion of collocation adopted in the paper. Collocation is a combination of two syntactically linked and semantically compatible content words, such as the pair's *main goal* and *moved with grace*. Syntactic links are realized directly or through an auxiliary word (usually a preposition). If any of conditions indicated above does not hold, the corresponding word combination is not collocation, for example, *the forest*, *river slowly*, *boiling goal*.

There are several syntactic types of collocations in each language. The most frequent types in European languages are: "the modified word → its modifier"; "noun → its noun complement"; "verb → its noun complement"; "verb predicate → its subject"; and "adjective → its noun complement". Directed links reflect syntactic dependency "head → its dependent".

The most frequent types and subtypes of Russian collocations are given in Table 1. They are determined by POS of collocatives and their order in texts; **N** symbolizes noun, **Adj** is adjective or participle, **V** and **Adv** are verb and adverb correspondingly, and **Pr** is preposition. Subindex *comp* means noun complement, while subindex *sub* means the noun subject in nominative case. Subindex *pred* symbolizes specifically Russian predicative short form of adjectival.

Table 1. Frequent types and structures of Russian collocations

Type title	Type code	Type structure	English example	Russian example
modified → its modifier	1.1	Adj ← N	*strong tea*	*крепкий чай*
	1.2	Adv ← Adj	*very good*	*очень хороший*
noun → its noun complement	2.1	N → N_{comp}	n/a	*огни города*
	2.2	N→ Prep → N_{comp}	*signs of life*	*вызов в суд*
verb → its noun complement	3.1	V → N_{comp}	*give message*	*искать решение*
	3.2	V → Prep → N_{comp}	*go to cinema*	*идти в кино*
	3.3	N_{comp} ← V	n/a	*здание затушили*
verb predicate → its subject	4.1	N_{sub} ← V	*light failed*	*мальчик пел*
	4.2	V → N_{sub}	*(there) exist people*	*пропали письма*
	4.3	Adj_{pred} → N_{sub}	n/a	*отправлен груз*
	4.4	N_{sub} ← Adj_{pred}	n/a	*порт открыт*
adjective → its noun complement	5.1	Adj → Prep → N_{comp}	*easy for girls*	*красный от стыда*
	5.3	Adj → N_{comp}	n/a	*занятый трудом*

Within a sentence, collocatives may be adjacent either distant from each other. The distribution of possible distances depends on the collocation type and specific collocatives. For example, collocatives of subtypes 2.1 and 2.2 are usually adjacent, whereas the 3.1-collocation such as *give → message* can contain intermediate contexts of lengths 0 to 4 and even longer, e.g. *give her a short personalized message*.

Our definition of collocations ignores their frequencies and idiomatically. As for frequencies, the advance of the Web shows that any semantically compatible word combination eventually realizes several times, thus we can consider as collocations all those exceeding a rather low threshold.

The main idea of our method of malapropism detection is to look through all pairs of content words within the sentence under revision, testing its syntactic links and its semantic admissibility. If the pair (*V, W*) is syntactically connected but semantically incompatible, a malapropism is signaled.

When a malapropism is detected, it is not known which collocative is erroneous, so we should try to correct both of them. The situation is clarified in Fig. 1. The upper two collocative nodes form the malapropism. The nodes going left-and-down and right-and-down are corresponding paronyms for malapropism's nodes. Each paronym should be matched against the opposite malapropism's node, and any pair may be admissible, but only one combination corresponds to the intended collocation; we call it *true correction*.

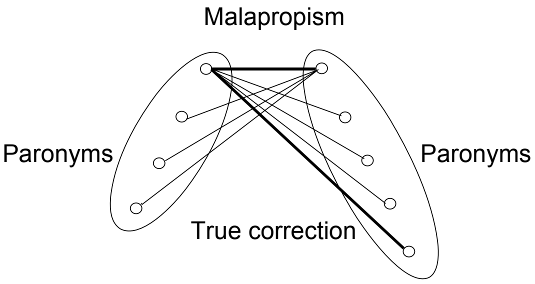


Fig. 1. Correction candidates and true correction

In such a way, all possible pairs of a collocative and its counterpart's paronym are formed, and we call them *primary candidates* for correction. The candidates are tested on semantic compatibility. If a pair fails, it is discarded; otherwise it is included into a list of *secondary candidates*. Then this list is ranked and only the best candidates are kept.

Obviously, for testing pairs (*V, W*) on semantic compatibility, using the Web as a text corpus, a statistical criterion is needed. According to one criterion, the pair is compatible if the relative frequency $N(V,W)/S$ of the co-occurrence of its words in a short distance in the whole corpus is greater than the product of relative frequencies $N(V)/S$ and $N(W)/S$ of occurrences of V and W taken separately (N means frequency; S is the size of the corpus). Using logarithms, we have the following threshold rule of pair compatibility:

$$\mathrm{MII}(V, W) \equiv \ln(N(V, W)) + \ln(S) - \ln(N(V)) - \ln(N(W) > 0,$$

where MII(*V, W*) is the mutual information index [7].

Since any search engine automatically delivers statistics about the queried word or the word combination measured in numbers of pages, to heuristically estimate the pair compatibility we propose a Semantic Compatibility Index (SCI) similar to MII:

$$\mathrm{SCI}(V,W) \equiv \begin{cases} \ln(P) + \ln(N(V,W)) - (\ln(N(V)) + \ln(N(W))) / 2, & \text{if } N(V,W) > 0, \\ NEG, & \text{if } N(V,W) = 0, \end{cases}$$

where N is the number of relevant pages; P is a positive constant to be chosen experimentally; and NEG is a negative constant. A merit of SCI as compared to MII is that the total number of pages is not to be estimated. Similarly to MII, SCI does not depend on monotonic or oscillating variations of all statistical data in the search engine because of the divisor 2.

If SCI(V_m, W_m) < 0, the pair (V_m, W_m) is malapropism, whereas the primary candidate (V, W) is selected as a secondary one according to the following threshold rule:

$$(\mathrm{SCI}(V_m,W_m) = NEG) \textbf{ and } (\mathrm{SCI}(V,W) > Q) \quad \textbf{or} \quad (\mathrm{SCI}(V_m,W_m) > NEG) \textbf{ and } (\mathrm{SCI}(V,W) > \mathrm{SCI}(V_m,W_m))$$

where Q ($NEG < Q < 0$) is a constant to be chosen experimentally.

The resulted set of secondary candidates is ranked by SCI values. The *best candidates* are all with positive SCI (let be *n* of them), whereas only one candidate with a negative SCI value is admitted, if *n*=1, and two candidates, if *n*=0.

Experimental Sets of Malapropisms

For experiments, we use two experimental sets – both of them consist of hundred of Russian sample malapropisms, which were mainly formed with the aid of the Web newswire. Specifically, we extracted collocations from the news messages, and one collocative of each collocation was then falsified using one of paronymy dictionaries, as a rule, the dictionary of literal paronyms (since literal errors are much more frequent in any language than morphemic ones). While falsifying, the morphological features of the word being changed (number, gender, person, case, etc.) were retained.

Then we again used paronymy dictionaries to make all possible correction candidates for each formed malapropism: through replacing of one word of the malapropism by its paronym we obtained the corresponding primary candidate. Among primary correction candidates, the before mentioned true correction (identical with intended collocation) was necessarily appeared.

1)1L 1.1 (проявил) кассовое сознание	'cash consciousness'
1L массовое сознание	'mass consciousness'
1L!! классовое сознание	'class consciousness'
1L кастовое сознание	'caste consciousness'
2L кассовое создание	'cash creature'
2M кассовое знание	'cash knowledge'
2M кассовое признание	'cash confession'
2M кассовое осознание	'cash perception'
2L кассовое познание	'cash cognition'
2)2L! 1.3 (песня)явно сдалась	'evidently capitulated'
2L!! явно удалась	'evidently succeed'
2M явно задалась	'evidently preset'
2M явно далась	'evidently given'
2M явно продалась	'evidently sold'
1L ясно сдалась	'clearly capitulated'
2M явно подалась	'evidently gone'
3)1L 2.1 (занят)смирением террористов	'by submission of terrorists'
1L!! усмирением террористов	'by pacification of terrorists'
1M примирением террористов	'by reconciliation of terrorists'
4)1L 2.2 кастеты с кадрами	'knuckledusters with frames'
1L!! кассеты с кадрами	'cassettes with frames'
2L кастеты с карами	'knuckledusters with retributions'
2L кастеты с кедрами	'knuckledusters with cedars'
1L катеты с кадрами	'legs with frames'
5)2L 3.3 протокол подманили	'protocol is dangled'
2L!! протокол подменили	'protocol is replaced'
2L протокол поманили	'protocol is drown on'
2L протокол подранили	'protocol is injured'

Fig. 2. Several malapropisms and their correction candidates with translations

The resulted sets consist of enumerated sample groups, each group corresponding to a malapropism and its primary candidates. Several sample groups are given in the first column of Fig. 2. Headlines of groups begin with the number of the changed collocative (1 or 2) and the symbol of the used paronymy dictionary (**L**iteral or

Morphemic). The next is code $n_1.n_2$ of syntactic type of the collocation (cf. Table 1), and then goes the malapropism string, may be with a short context given in parentheses. Lines with correction candidates begin with the number of the changed word (1 or 2) and the symbol of the used paronymy dictionary; true corrections are marked with ‘**!!**’. The translation of malapropisms and their correction variants in the second column of Fig. 2. exhibits the nonsense of wrong corrections.

In total, the first malapropism set includes 648 primary correction candidates, and the second, 737 candidates. So the mean number of primary correction candidates is ≈ 7 candidates per error.

Among the samples, the sets include also errors named *quasi-malapropisms* (their total number equals 16). A quasi-malapropism transforms one collocation to another semantically legal collocation, which can be rarer and contradict to the outer context, e.g., *normal manner* changed to *normal banner* or *give message* changed to *give massage*. An example of Russian *quasi-malapropism* is presented in Fig. 2, it is marked with ‘**!**’ (cf. the second sample group). The detection of quasi-malapropisms (if possible) sometimes permits one to restore the intended words, just as for malapropisms proper.

Experiments with Yandex and Their Results

A specific collocation or malapropism met in a text has its certain distance between collocatives. However, to reliably detect malapropisms by means of the Web and the selected statistic criterion, we should put each word pair being tested in its most probable distance.

For this reason, we initially explored frequencies of various Russian collocative co-occurrences against the distance between them on the base of Yandex statistics, cf. Table 2 and Table 3. The statistics of co-occurrence frequencies (measured in the number of relevant pages) were accumulated for twelve collocations of various frequent types. The used queries contained collocatives in quotation marks separated with */n* indicating the distance *n* between the given words, for example, *+“столбы”/2+“дыма”*. Such queries give frequencies of the words encountered within the same sentence with distance between them equal to *n* (or the number of intermediate words equals to *n-1*).

The statistics show that, for all collocations, frequency maximums correspond to the numbers 0 or 1 of intermediate words, and such cases cover more than 60% of encountered word pairs (cf. the last column of Table 2). Since we cannot determine automatically whether counted Web co-occurrences are real collocations or mere encounters of words without direct syntactic links between, we look through the first fifty page headers, mentally analyzing their syntax. Thereby we ascertained that the most of the co-occurrences with adjacent collocatives or those separated by one word are real collocations.

Table 2. Yandex statistics of collocative co-occurrences

Collocation	Type	Number of intermediate words:					Percents in 0 and 1
		0	1	2	3	4	
Уделить внимание	3.1	52248	**72433**	9111	3537	1335	90%
Отправлен груз	4.3	779	**3408**	100	17	8	97%
Сбор информации	2.1	**141395**	32342	54354	31326	13566	64%
Спасатели обнаружили	4.1	**18534**	2440	929	524	740	91%
Здание потушили	3.3	**48**	7	14	10	0	70%
Сроки рассмотрения	2.1	**31517**	2918	2302	891	1075	89%
Затонувшее судно	1.1	**10250**	496	189	642	128	92%
Оценка деятельности	2.1	**29276**	22847	20373	5370	4183	64%
Занятый трудом	5.3	40	**413**	215	16	11	65%
Столбы дыма	2.1	**4382**	1420	507	79	93	90%
Сделать оговорки	3.1	355	**660**	269	44	14	76%
Приведем пример	3.1	**30665**	13106	6343	1376	580	84%

Thus, we can deduce that for frequent types of Russian collocations the most probable distance between collocatives (measured in the number of intermediate words) equals 0 or 1. As to collocatives linked through prepositions, the most probable distances at both interval are equal to 0, cf. Table 3. The table shows the distribution of frequencies for possible combinations of two distances: between the first collocative and the preposition and the preposition and the second collocative (e.g., combination 0-1 means that the first collocative and the preposition are adjacent, whereas the preposition and the second collocative are separated by one word).

Table 3. Yandex statistics of co-occurrences for collocations with prepositions

Collocation	Number of intermediate words									
	0 –0	1–0	0–1	2–0	1–1	0–2	3–0	2–1	1–2	0–3
ворвались в здание	**9869**	123	156	69	0	74	24	1	9	2
тайники с оружием	**4775**	1	43	10	1	29	3	0	0	38
вызов в суд	**3633**	281	91	90	4	15	120	6	0	15
справиться с управлением	**5744**	16	177	2	0	11	2	0	0	12

Then we have applied our method to the both experimental sets by means of the computer program that gathers statistics of word pairs co-occurrences with the distance between them (measured in the number of intermediate words) equal to 0 or 1 (collocatives linked through a preposition were tested as adjacent triples). This in no way means that collocations cannot have more distant collocatives, but the Web is not suited for collocation testing at greater distances. The frequencies of word occurrences and co-occurrences gathered for several malapropisms are given in the second column of Fig. 3 (the repeating data for the collocatives are omitted).

```
1)1L 1.1 кассовое сознание       2, кассовое:354955,сознание:4770500
  1L массовое сознание           32973, массовое:916455
  1L!! классовое сознание        2927, классовое:38924
  1L кастовое сознание           56, кастовое:11799
  2L кассовое создание           10, создание:32199807
  2M кассовое знание             1, знание:7120311
  2M кассовое признание          0, признание:2437390
  2M кассовое осознание          0, осознание:823650
  2L кассовое познание           0, познание:605134
2)2L! 1.3 явно сдалась           13, явно:9871866, сдалась:198061
  2L!! явно удалась              6703, удалась:610646
  2M явно задалась               386, задалась:46599
  2M явно далась                 38, далась:88177
  2M явно продалась              2, продалась:24594
  1L ясно сдалась                2, ясно:10816398
  2M явно подалась               0, подалась:298216
3)1L 2.1 смирением террористов   0, смирением:79063,террористов:2762914
  1L!! усмирением террористов    3, усмирением:1787
  1M примирением террористов     0, примирением:17515
4)1L 2.2 кастеты с кадрами       0, кастеты:42266,кадрами:481878
  1L!! кассеты с кадрами         21, кассеты:2923258
  2L кастеты с карами            0, карами:19351
  2L кастеты с кедрами           0, кедрами:5666
  1L катеты с кадрами            0, катеты: 3151
5)2L 3.3 протокол подманили      0, протокол:7635243, подманили:3521
  2L!! протокол подменили        36, подменили:86957
  2L протокол поманили           0, поманили:7545
  2L протокол подранили          0, подранили:946
```

Fig. 3. Several malapropisms and their correction candidates with Yandex statistics

We used the first experimental set to adjust the necessary constants of our method. To obtain all negative SCI values for all proper malapropisms from the first set, we take *P* = 1200. The constant *NEG* = –100 is taken lower than SCI values of all occurrences counted as non-zero events. The constant *Q* = –7.5 is adjusted so that all candidates with non-zero occurrences have SCI values greater then this threshold.

Though all eight quasi-malapropisms were excluded while selecting the constant *P*, our method detects seven of them as malapropisms proper: their SCI values proved to be too low to be acknowledged as collocations. Our program selects 169 secondary candidates from 648 primary ones and then reduces them to 141 best correction candidates. Among the best candidates for the 99 malapropisms signaled, as many as 98 have true correction options, and only two of them are not first-ranked.

While testing the method and the determined constants on the second experimental malapropisms set, all its malapropisms and even all eight quasi-malapropisms were detected. 165 secondary candidates were selected from 737 primary ones; and the secondary ones were reduced to 138 best candidates. But for five detected malapropisms their true corrections do not enter corresponding lists of the best candidates, and three true corrections among them were not selected as secondary candidates (we admit that these collocations are rather infrequent in texts).

```
1)1L 1.1 кассовое сознание         -6,29  Detected
  1L массовое сознание              2,95  Best
  1L!! классовое сознание           2,11  Best
2)2L! 1.3 явно сдалась          -4,49     Detected
  2L!! явно удалась                 1,20  Best
  2M явно задалась                 -0,37  Best
3)1L 2.1 смирением террористов   -100,00  Detected
  1L!! усмирением террористов      -2,96  Best
4)1L 2.2 кастеты с кадрами       -100,00  Detected
  1L!! кассеты с кадрами           -6,28  Best
5)2L 3.3 протокол подманили      -100,00  Detected
  2L!! протокол подменили          -2,93  Best
```

Fig. 4. Several malapropisms and the best candidates with their SCI values

We should note that the occasional omission of a true correction does not seem too dangerous, since the user can restore it in the case of error detection. Nevertheless, the most commonly used collocations among primary correction candidates always enter into the list of the best candidates, as true corrections or not.

For both experiments, the lists of the best candidates contain 1 to 4 entries, usually 1 or 2 entries; cf. the detected malapropisms with corresponding SCI values and decision qualifications in Fig. 4. The total decrease of correction candidates, from the primary to the best, exceeds 5.

Hence the results of our experiments are rather promising: for our experimental sets, the method of testing semantic compatibility through the Web has the recall 0.995. The proposed SCI is a quite good measure for detecting malapropisms, and the proposed heuristic rule for selection of secondary correction candidates is appropriate, whereas the heuristic rule for selection of best candidates may be slightly improved.

Conclusions and Further Work

A method is proposed for automatic detection and computer-aided correction of malapropisms. Experimental justification of the method was done on two representative sets of Russian malapropisms with the aid of Yandex search engine. While testing word pairs on their semantic compatibility through the Web, the most probable distances between the Russian words were taken into account.

Since the experiments gave good results, the problem of the Web statistics validity for collocation testing deserves to be investigated deeper. It would be worthwhile to extend the results of our study to broader experimental data and to other Web search engines. Of course, it is quite topical to develop a local grammar parser appropriate for malapropism detection, since for our experiments we extracted collocation components manually.

Since the Web proved to be adequate for testing semantic compatibility of collocations, we hope to use the method to develop procedures of automatic acquisition of collocation databases.

Bibliography

1. Bolshakov, I.A. Getting One's First Million…Collocations. In: A. Gelbukh (Ed.). Computational Linguistics and Intelligent Text Processing. Proc. 5th Int. Conf. on Computational Linguistics CICLing-2004, Seoul, Korea, February 2004. LNCS 2945, Springer, 2004, p. 229-242.
2. Bolshakov, I.A., A. Gelbukh. Paronyms for Accelerated Correction of Semantic Errors. International Journal on Information Theories & Applications. V. 10, N 2, 2003, p. 198-204.
3. Bolshakov, I.A., A. Gelbukh. On Detection of Malapropisms by Multistage Collocation Testing. In: A. Düsterhöft, B. Talheim (Eds.) Proc. 8th Int. Conference on Applications of Natural Language to Information Systems NLDB´2003, June 2003, Burg, Germany, GI-Edition, LNI, V. P-29, Bonn, 2003, p. 28-41.
4. Bolshakova, E.I. Towards Computer-aided Editing of Scientific and Technical Texts. International Journal on Information Theories & Applications. V. 10, N 2, 2003, p. 204-210.
5. Hirst, G., D. St-Onge. Lexical Chains as Representation of Context for Detection and Corrections of Malapropisms. In: C. Fellbaum (ed.) WordNet: An Electronic Lexical Database. MIT Press, 1998, p. 305-332.
6. Kilgarriff, A., G. Grefenstette. Introduction to the Special Issue on the Web as Corpus. Computational linguistics, V. 29, No. 3, 2003, p. 333-347.
7. Manning, Ch. D., H. Schütze. Foundations of Statistical Natural Language Processing. MIT Press, 1999.

Authors' Information

Elena I. Bolshakova – Moscow State Lomonosov University, Faculty of Computational Mathematics and Cybernetic, Algorithmic Language Department; Leninskie Gory, Moscow State University, VMK, Moscow 119899, Russia; e-mail: bolsh@cs.msu.su

Igor A. Bolshakov – Center for Computing Research (CIC), National Polytechnic Institute (IPN); Av. Juan Dios Bátiz esq. Av. Miguel Othon Mendizabal s/n, U.P. Adolfo Lopez Mateos, Col. Zacatenco, C.P. 07738, Mexico D.F., Mexico; e-mail: igor@cic.ipn.mx

Alexey P. Kotlyarov – Moscow State Lomonosov University, Faculty of Computational Mathematics and Cybernetic, Algorithmic Language Department; Leninskie Gory, Moscow State University, VMK, Moscow 119899, Russia; e-mail: koterpillar@yandex.ru

A MATHEMATICAL APPARATUS FOR DOMAIN ONTOLOGY SIMULATION. AN EXTENDABLE LANGUAGE OF APPLIED LOGIC[1]

Alexander Kleshchev, Irene Artemjeva

Abstract: *A mathematical apparatus for domain ontology simulation will be described in the series of the articles. This article is the first one of the series. The paper is devoted to means for representation of domain models and domain ontology models, so here a logical language is used only as a means for formalizing ideas. The chief requirement to such a language is that it must have such a semantic basis that would allow us to determine the most exact approximation of a set of intended interpretation functions as often as possible. Another requirement closely connected with the foregoing one is that the awkwardness of expressing ideas in such a language must not considerably exceed the complexity of their expressing in natural language. There are two ways to meet the requirements. The first one is to define and fix a wide semantic basis of the language. In this case the semantic basis nonetheless can be insufficient for some applications of the language. Extending applications of the language can lead from time to time to the necessity of further extending its semantic basis, i.e. to the*

[1] This paper was made according to the program of fundamental scientific research of the Presidium of the Russian Academy of Sciences «Mathematical simulation and intellectual systems», the project "Theoretical foundation of the intellectual systems based on ontologies for intellectual support of scientific researches".

necessity of defining new and new versions of the language. The second way is to make the kernel of the language being as nearer to the semantic basis of the classical language as possible and to allow us to make necessary extensions of the kernel for particular applications. In this article the second way is used to define the extendable language of applied logic. The goal of this article is to define the kernel of the extendable language of applied logic and its standard extension. The standard extension of the language defines elements of the semantic basis that are supposed to be useful practically in all the applications.

Keywords: Extendable language of applied logic, ontology language specification, kernel of extendable language of applied logic, the standard extension of the language of applied logic.

ACM Classification Keywords :I.2.4 Knowledge Representation Formalisms and Methods, F4.1. Mathematical Logic

Introduction

At present the importance of studying properties of domain ontologies is generally recognized [Guarino, 1998] [Studer et al, 1998]. As a rule, a language of predicate calculus of the first order, other languages of mathematical logic [Guarino, 1998] [Wielinga et al, 1994] or ontology description languages [van Heijst et al, 1996] are used for formal representation of ontologies.

Languages of mathematical logic were created for aims that were not connected directly with describing domain ontologies. Therefore, they do not allow us to represent formally all the properties of domain ontologies. In particular, the definition of values and sorts for names of a signature, distinctions between the propositions representing domain knowledge and ontological agreements are beyond the syntax and semantics of these languages. Some poorness of the semantic basis for the most of such languages leads to the fact that the meaning of ontological agreements is obscured by awkwardness of technical details which are necessary to express the meaning in these languages. In addition, the languages of the first order do not allow us to introduce terms of a high level of generality and not taking part in description of situations (states of affairs in terms of the paper [Guarino, 1998]) into ontology descriptions. As a consequence, an ontology description representing properties of all domain terms turns out immense even if these terms can be divided into classes of terms with the same properties.

Semantics of ontology description languages and of predicate calculus languages is equivalent. Both ontologies and domain models can be described in these languages. Therefore, it is not clear in what measure such languages are specific for formalizing exactly domain ontologies and in what way this specificity is represented by their syntax and semantics. Operational aspects of semantics of these languages are not connected with domain ontologies but can be important only for describing task and method ontologies.

In papers [Artemjeva et al, 1995, 1996, 1997a, 1997b], [Kleshchev et al, 1998] an attempt was made to suggest a mathematical apparatus - logic relationship systems - for domain simulation. This apparatus cleared up some of the above troubles. But direct application of this apparatus for simulation of domain ontologies is impossible because domain ontology models are sets of domain models. For such an application it is necessary to make an appropriate generalization of the apparatus.

The goal of the article series is to build a mathematical apparatus for domain ontology simulation. In the article an extendable language of applied logic is defined and also the standard extension of the language. This language will be used for representation of mathematical models of domain ontologies.

1. The Necessity of an Extendable Language of Applied Logic

In this article an extendable language of applied logic is defined. The necessity of the language is motivated by the following circumstances. The language of classical mathematical logic - the language of predicate calculus - was developed to attain two aims conflicting with one another in many respects. First, it is a means for description of generative process states in a predicate calculus (of sets of formulas). Second, it is a means for formalizing ideas.

Any language for description of generative process states (of sets of formulas) in a predicate calculus has to have such a semantic basis (a set of language symbols whose semantics does not depend on an interpretation of the signature symbols) that the predicate calculus being complete contains a finite set of inference rules. The semantic basis of the classical language of predicate calculus is formed by propositional connectives and quantifiers. In addition, sets (domains of variable values), Cartesian product of sets (domains of definitions of functions and predicates are Cartesian products) and also functional maps (the interpretations of functional and predicate symbols are elements of the functional map set) belong to the semantic basis implicitly. The inference rules of the predicate calculus usually correspond to the semantics of propositional connectives and quantifiers.

With the use of a predicate calculus language as a means for formalizing ideas, every logic theory can be considered as a way of intensional determining a set of interpretation functions having the same finite domain of definition - the signature of the language. In this case, the language has to have such a semantic basis that the most exact approximation of any intended set of interpretation functions can be determined by a finite set of propositions. It is clear, the wider such a semantic basis, the more often this aim can be achieved. In real applications, when declarative models (systems of algebraic, differential and other equations, and also systems of inequalities and optimization tasks) are determined this semantic basis contains arithmetic at the minimum. However, it is well known that a predicate calculus being complete and based on such a language cannot contain a finite set of inference rules.

In this manner, there are, as they are, a few logic languages. Some of them (for describing formulas) have a restricted semantic basis but the others (for formalizing ideas) have an extendable semantic basis. The languages of the first group are means for representation of generative process states in predicate calculus studied within mathematical logic. But the others are means for representation of declarative models studied within abstract and applied mathematics.

This paper is devoted to means for representation of domain models and domain ontology models, so here a logic language is used only as a means for formalizing ideas. The chief requirement to such a language is that it must have such a semantic basis that would allow us to determine the most exact approximation of a set of intended interpretation functions as often as possible. Another requirement closely connected with the foregoing one is that the awkwardness of expressing ideas in such a language must not considerably exceed the complexity of their expressing in natural language. There are two ways to meet the requirements. The first one is to define and fix a wide semantic basis of the language. In this case the semantic basis nonetheless can be insufficient for some applications of the language. Extending applications of the language can lead from time to time to the necessity of further extending its semantic basis, i.e. to the necessity of defining new and new versions of the language. The second way is to make the kernel of the language being as nearer to the semantic basis of the classical language as possible and to allow us to make necessary extensions of the kernel for particular applications.

In this article the second way is used to define the language of applied logic. The definition of the language consists of the kernel of the language only. When the semantic basis is extended for particular applications the following two classes of elements are possible. The elements of the first class can be impossible or undesirable to be defined by means of the kernel of the language and by extensions built. On the contrary, the elements of the second class can be naturally defined by means of the kernel and extensions built. The elements of the first class are described in the standard extension and in specialized extensions in the same form that is used in the description of the kernel of the language. The standard extension of the language defines elements of the semantic basis that are supposed to be useful practically in all the applications. A specialized extension of the language defines elements of the semantic basis that are necessary for a comparatively narrow class of applications. Because the same specialized extensions can be used in different applications such extensions have names. Every particular language of applied logic contains the kernel and usually the standard extension and possibly some specialized extensions. By this means, every particular language of applied logic is characterized by a set of extension names rather than a signature. A signature is introduced by a particular logical theory represented in such a language. Therewith, propositions of the theory can associate values (interpretation) or sorts with names (elements of the signature) or can restrict possible functions of interpretations for these names according to the interpretation of other names. In turn, every theory has a name. The parameters of the name are the names of the extensions of the language that are used for describing the theory. Other theories represented by their names also can be elements of a theory.

In this article the syntax and semantics of auxiliary constructions of the language (terms and formulas) and its basic constructions (propositions and logic theories) are defined.

2. The Kernel of the Applied Logic Language. The Syntax of Terms, Formulas, Propositions and Applied Logic Theory

The syntax of terms. The terms are:

1. a name n;
2. a variable v;
3. N and L;
4. $t_1 \rightarrow t_2$, where t_1 and t_2 are terms;
5. $(\times\ t_1,..., t_k)$, where $t_1, ..., t_k$ are terms;
6. $t(t_1, ..., t_k)$, where t , $t_1, ..., t_k$ are terms;
7. j(t), where t is a term.

The syntax of formulas. The formulas are:

1. $t(t_1, ..., t_k)$, where t , $t_1, ..., t_k$ are terms;
2. $\neg f_1$, f_1 & f_2, $f_1 \vee f_2$, $f_1 \Rightarrow f_2$, $f_1 \Leftrightarrow f_2$, where f_1 and f_2 are formulas.

If a variable is not bound in a term or in a formula then it is considered as free in the term or in the formula. If a variable is bound in a term or in a formula then it is bound also in the proposition including this term or this formula. There are no bound variables in the kernel of the language.

The syntax of propositions. A proposition consists of a prefix and a body. A prefix is a set of variable descriptions $(v_1: t_1)...(v_m: t_m)$ (bounded universal quantifiers), where $(v_i: t_i)$ is a variable description, v_i is a variable, t_i is a term for all i=1, ...,m. For I = 1,...,m only the variables $v_1, ..., v_m$ can be free variables of the term terms $t_1, ..., t_m$. A set of variable descriptions can be empty. All the variables $v_1, ..., v_m$ are mutually different.

The body of a proposition depends on the type of the proposition. The types of propositions are a value description for a name, a sort description for a name, a restriction on the interpretation of names. Any free variable which is a part of the body of a proposition must be described in its prefix. If a variable is bound in the body of a proposition then it cannot be a part of the prefix of the proposition.

The body of a value description for a name has a form $t_1 \equiv t_2$, where t_1 and t_2 are terms.

The body of a sort description for a name has a form sort $t_1 : t_2$, where t_1 and t_2 are terms.

The body of a restriction on the interpretation of names is a formula.

The syntax of applied logic theories. An applied logic theory named $T(E_1, ..., E_k)$, where $E_1, ..., E_k$ are the names of extensions of the language used for representing the theory is a pair <TS, SS>, where TS is a finite set (perhaps empty) of names of other theories, SS is a finite set (perhaps empty) of propositions. Any applied logic theory T = <TS, SS> by definition is equivalent to an applied logic theory <∅, SS'>, where SS' is the result of the following process. Let us denote ts(T) = TS, ss(T) =SS. Let TS_1= ts(T) and SS_1= ss(T). For every i = 1,2, ... let $TS_{i+1} = \bigcup_{t \in TS_i} ts(t)$, $SS_{i+1} = SS_i \cup \bigcup_{t \in TS_i} ss(t)$. If $TS_n = \varnothing$ on a recurrent step n then SS' = SS_n. The theory <∅, SS'> will be called the reduction of the theory <TS, SS>.

An applied logic theory will be called *syntactically correct* if

- TS contains only syntactically correct applied logic theories;
- SS contains propositions written by means of the kernel of the language and of its extensions $E_1, E_2, ..., E_k$ only;
- the above process for building the reduction of the logic theory is completed in a finite number of steps;
- the reduction of the theory T contains a nonempty set of propositions.

It is evident that the set of propositions of the reduction of any syntactically correct applied logical theory is a finite set.

3. The Kernel of the Applied Logic Language. Semantics of Terms, Formulas, Propositions and Applied Logic Theory

Semantics of terms and formulas. Semantics of terms and formulas determines the values of terms and formulas and also the conditions under which these values exist. In this case it is suggested that a function α is given on the set of names. For every name the value of the function is an interpretation of the name. The values of terms and formulas will be defined in relation to an interpretation function α and an arbitrary admissible substitution θ of values for all the free variables in the term or in the formula. If a variable being free in a term or in a formula is also free in the proposition including the term or the formula then in an admissible substitution θ the value for the variable is determined by the semantics of the proposition. But if a variable being free in a term or in a formula is bound in the proposition including the term or the formula then in an admissible substitution θ the value for the variable is determined by the semantics of the term or of the formula in that the variable is bound. Let $J_{\alpha\theta}(t)$ denote the value of a term t for an interpretation function α and an admissible substitution θ, $J_{\alpha\theta}(f)$ denote the value of a formula f for an interpretation function α and an admissible θ, $\theta(v)$ denote the value of a variable v in the substitution θ.

The values of terms are defined by the following way.

1. $J_{\alpha\theta}(n) = \alpha(n)$, where n is a name; $J_{\alpha\theta}(n)$ does not depend on θ; the value $J_{\alpha\theta}(n)$ exists if n is an element of the set $J_{\alpha\theta}(N)$;

2. $J_{\alpha\theta}(v) = \theta(v)$, where v is a variable;

3. $J_{\alpha\theta}(N)$ is the infinite set of all possible names; $J_{\alpha\theta}(N)$ does not contain all the names that are described in the standard and in any used specialized extension of the language and also "N", "L", "≡","=", "→", "×", "⇒", "∨", "&", "¬", "⇔", "(", ")", ":", "true", "false", ",", "sort", "j"; $J_{\alpha\theta}(N)$ does not depend on α and θ;

4. $J_{\alpha\theta}(L)$ is the set consisting of two elements true and false; $J_{\alpha\theta}(L)$ does not depend on α and θ;

5. $J_{\alpha\theta}(t_1 \rightarrow t_2)$ is the set of all possible completely defined functions from the set $J_{\alpha\theta}(t_1)$ to the set $J_{\alpha\theta}(t_2)$; the value of the term exists if the both values $J_{\alpha\theta}(t_1)$ and $J_{\alpha\theta}(t_2)$ are sets;

6. $J_{\alpha\theta}(\times\, t_1, ..., t_k)$ is the Cartesian product of the sets $J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)$; the value of the term exists if all the values $J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)$ are sets; the operation "×" has all the properties of Cartesian product but associativity $J_{\alpha\theta}(\times(\times\, t_1, t_2), t_3) \neq J_{\alpha\theta}(\times\, t_1, (\times\, t_2, t_3))$;

7. $J_{\alpha\theta}(t(t_1, ..., t_k)) = \varphi(J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k))$ is the value of the function φ which is the interpretation of the name $J_{\alpha\theta}(t)$ (i.e. $\varphi = \alpha(J_{\alpha\theta}(t))$), applied to the arguments $J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)$; the value of the term exists if the value $J_{\alpha\theta}(t)$ is a name, having a sort $(s' \rightarrow s)$, where s' is the Cartesian product of the sets $s_1, ..., s_k$ or a subset of the Cartesian product, s is a set, with $s \neq J_{\alpha\theta}(L)$, $<J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)> \in s'$; in this case $J_{\alpha\theta}(t(t_1, ..., t_k)) \in s$; let us notice that if t' is such a term that $J_{\alpha\theta}(t') = <J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)>$ then $J_{\alpha\theta}(t(t')) = J_{\alpha\theta}(t(t_1, ..., t_k))$;

8. $J_{\alpha\theta}(j(t)) = \alpha(J_{\alpha\theta}(t))$ is the interpretation of the name $J_{\alpha\theta}(t)$; the value of the term exists if $J_{\alpha\theta}(t)$ is a name.

The values of formulas are defined in the following way.

1. $J_{\alpha\theta}(t(t_1, ..., t_k)) \Leftrightarrow \rho(J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k))$ is the value of the predicate ρ, which is the interpretation of the name $J_{\alpha\theta}(t)$ (i.e. $\rho = \alpha(J_{\alpha\theta}(t))$) applied to the arguments $J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)$; the formula has a value if the value $J_{\alpha\theta}(t)$ is a name having a sort $(s' \rightarrow L)$, where s' is the Cartesian product of the sets $s_1, ..., s_k$ or a subset of the Cartesian product, $<J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)> \in s'$; let us notice that if t' is such a term that $J_{\alpha\theta}(t') = <J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)>$ then $J_{\alpha\theta}(t(t')) \Leftrightarrow J_{\alpha\theta}(t(t_1, ..., t_k))$;

2. $J_{\alpha\theta}(\neg\, f) \Leftrightarrow \neg\, J_{\alpha\theta}(f)$, i.e. the value of the formula $\neg$ f is true if and only if the value $J_{\alpha\theta}(f)$ is false; the formula has a value if the formula f has a value for the interpretation function α and the substitution θ;

3. $J_{\alpha\theta}(f_1 \,\&\, f_2) \Leftrightarrow J_{\alpha\theta}(f_1) \,\&\, J_{\alpha\theta}(f_2)$, i.e. the value of the formula f_1 & f_2 is true if and only if the both values $J_{\alpha\theta}(f_1)$ and $J_{\alpha\theta}(f_2)$ are true; the formula has a value if the both formulas f_1 and f_2 have values for the interpretation function α and the substitution θ;

4. $J_{\alpha\theta}(f_1 \vee f_2) \Leftrightarrow J_{\alpha\theta}(f_1) \vee J_{\alpha\theta}(f_2)$, i.e. the value of the formula $f_1 \vee f_2$ is true if and only if at least one of the values $J_{\alpha\theta}(f_1)$ or $J_{\alpha\theta}(f_2)$ is true; the formula has a value if the both formulas f_1 and f_2 have values for the interpretation function α and the substitution θ;

5. $J_{\alpha\theta}(f_1 \Rightarrow f_2) \Leftrightarrow J_{\alpha\theta}(f_1) \Rightarrow J_{\alpha\theta}(f_2)$, i.e. the value of the formula $f_1 \Rightarrow f_2$ is true if and only if either the value $J_{\alpha\theta}(f_1)$ is false or the both values $J_{\alpha\theta}(f_1)$ and $J_{\alpha\theta}(f_2)$ are true; the formula has a value if the both formulas f_1 and f_2 have values for the interpretation function α and the substitution θ;

6. $J_{\alpha\theta}(f_1 \Leftrightarrow f_2) \Leftrightarrow J_{\alpha\theta}(f_1) \Leftrightarrow J_{\alpha\theta}(f_2)$, i.e. the value of the formula $f_1 \Leftrightarrow f_2$ is true if and only if either the both values $J_{\alpha\theta}(f_1)$ and $J_{\alpha\theta}(f_2)$ are false or the both values $J_{\alpha\theta}(f_1)$ and $J_{\alpha\theta}(f_2)$ are true; the formula has a value if the both formulas f_1 and f_2 have values for the interpretation function α and the substitution θ;

Semantics of propositions. Semantics of propositions determines the meaning of the propositions and also the conditions under which propositions have meaning.

The set of admissible substitutions θ for free variables of a proposition is formed in the following way. If the prefix of the proposition is empty then the set of admissible substitutions of the proposition consists of the only empty substitution. Let the prefix of the proposition be of the form $(v_1\text{: } t_1)...(v_m\text{: } t_m)$, then the set of admissible substitutions is the set of all the substitutions $\theta = (v_1/c_1, ..., v_m/c_m)$, where $c_1 \in J_{\alpha\theta 1}(t_1), ..., c_m \in J_{\alpha\theta m}(t_m)$.

A value description for a name with the body $t_1 \equiv t_2$ has the following meaning: for every admissible substitution θ the interpretation of the name $J_{\alpha\theta}(t_1)$ is $J_{\alpha\theta}(t_2)$. The proposition has meaning if for all the admissible substitutions the value $J_{\alpha\theta}(t_1)$ is a name, the value of the term t_2 exists for the interpretation function α and for the substitution θ and also it does not follow from the logical theory that the name $J_{\alpha\theta}(t_1)$ has more than one value. A set of value descriptions for names can contain recursive value definitions for names.

A sort description for a name with the body sort $t_1 : t_2$ has the following meaning: for every admissible substitution θ the name $J_{\alpha\theta}(t_1)$ has the sort $J_{\alpha\theta}(t_2)$. The proposition has meaning if for all the admissible substitutions $J_{\alpha\theta}(t_1)$ is a name, $J_{\alpha\theta}(t_2)$ is a set and it does not follow from the logical theory that the name $J_{\alpha\theta}(t_1)$ has more than one sort. A set of sort descriptions for names can contain recursive sort definitions for names.

If a sort description for a name has the body sort $t_1 : t_2 \to t_3$ and $J_{\alpha\theta}(t_3) \neq J_{\alpha\theta}(L)$ then we will say that the name $J_{\alpha\theta}(t_1)$ is a functional name; if $J_{\alpha\theta}(t_3) = J_{\alpha\theta}(L)$ then we will say that the name $J_{\alpha\theta}(t_1)$ is a predicative name; otherwise we will say that the name $J_{\alpha\theta}(t_1)$ is an objective name.

A restriction on the interpretation of names has the following meaning: an interpretation function α is admissible if $J_{\alpha\theta}(f)$ = true for all the admissible substitutions θ, where f is a formula that is the body of this proposition. The proposition has meaning if there is such an interpretation function that the formula f is true for all the admissible substitutions θ.

Semantics of applied logic theories. The set of names being parts of an applied logic theory can be divided into two nonintersecting subsets: a set of uniquely interpreted names and a set of ambiguously interpreted names. A name is uniquely interpreted if one of the following conditions is met:

- the applied logic theory determines neither any sort nor any value for a name n; in this case for any α the interpretation $\alpha(n) = n$;

- the applied logic theory determines a value e for a name n and the value does not depend on the interpretations of other names; in this case for any α the interpretation $\alpha(n) = e$;

- the applied logic theory determines a value e for a name n and the value is uniquely determined by the interpretations of other names.

All the other names are ambiguously interpreted. For every such a name the applied logic theory determines a sort s but does not determine any value. In this case any interpretation function α must meet the restriction $\alpha(n) \in s$.

An interpretation function α is admissible for an applied logic theory if all the propositions of the theory reduction have meaning for this interpretation function. An applied logic theory is semantically correct if there is an admissible interpretation function α. Since for every proposition the set of admissible substitutions is determined uniquely but the admissible interpretation function is determined ambiguously then a semantically correct applied logic theory determines a set of admissible interpretation functions. It is easily seen that under these conditions the set of ambiguously interpreted names of any semantically correct applied logic theory is finite for any admissible interpretation function.

The constriction of an admissible interpretation function α to the set of ambiguously interpreted names of an applied logic theory will be called a model of the theory. A model of an applied logic theory can be represented by such a set of value descriptions for names that after adding the set to the theory all the names of the new theory built in such a way will be uniquely interpreted.

4. The Standard Extension of the Language of Applied Logic

An extension of the language is a description of syntax and semantics for terms and formulas. These terms and formulas are added to the kernel of the language of applied logic. The standard extension ST of the language of applied logic introduces syntactic constructions for some special languages of mathematical logic and also arithmetic and set-theoretic constants, operations and relations. The syntax of many constructions is usual for mathematical expressions. All the quantifier constructions have a unified syntax $\mathrm{A}(v_1\colon t_1)\ldots(v_m\colon t_m)\ t\ \Omega$ or $\mathrm{A}(v_1\colon t_1)\ldots(v_m\colon t_m)\ f\ \Omega$, where A and Ω are quantifier brackets (unique for every quantifier), $(v_1\colon t_1)\ldots(v_m\colon t_m)$ is a set of variable descriptions, t is a term, and f is a formula. The variables $v_1, \ldots, v_m$ are bound in this construction.

The terms are:

1. a quantifier construction $(\iota(v_1\colon t_1)\ldots(v_m\colon t_m)\ f)$ (iota-operator); $J_{\alpha,\theta}((\iota(v_1\colon t_1)\ldots(v_m\colon t_m)\ f))$ is equal to such an element of the set of admissible substitutions for $(v_1\colon t_1)\ldots(v_m\colon t_m)$ that if it was substituted for the variables $v_1,\ldots, v_m$ then the value $J_{\alpha\theta}(f)$ would be true; the value of the term exists if the values $J_{\alpha\theta}(t_1), \ldots, J_{\alpha\theta}(t_m)$ are sets and such an element of the Cartesian product is unique that if it was substituted for the variables then the value $J_{\alpha,\theta}(f)$ would be true;

2. a quantifier construction $(\lambda(v_1\colon t_1)\ldots(v_m\colon t_m)\ t)$ (lambda-term determining a function); $J_{\alpha\theta}((\lambda(v_1\colon t_1)\ldots(v_m\colon t_m)\ t))$ is a function φ of m arguments; for any element $<q_1, .., q_m>$ of the set of admissible substitutions for $(v_1\colon t_1)\ldots\ (v_m\colon t_m)$ the value $\varphi(q_1,q_2,..,q_m) = J_{\alpha\theta}(t)$;

3. a quantifier construction $(\lambda(v_1\colon t_1)\ldots(v_m\colon t_m)\ f)$ (lambda-term determining a predicate); $J_{\alpha\theta}((\lambda(v_1{:}t_1)\ldots(v_m{:}t_m)\ f))$ is a predicate ρ of m arguments; for any element $<q_1,q_2,..,q_m>$ of the set of admissible substitutions for $(v_1\colon t_1)\ldots\ (v_m\colon t_m)$ the value $\rho(q_1,q_2,..,q_m) \Leftrightarrow J_{\alpha\theta}(f)$;

4. $/(f_1 \Rightarrow t_1), ..., (f_m \Rightarrow t_m)/$ (conditional term), where $t_1, .., t_m$ are terms and $f_1, .., f_m$ are formulas; $J_{\alpha\theta}(/(f_1 \Rightarrow t_1), \ldots, f_m \Rightarrow t_m)/) = J_{\alpha\theta}(t_k)$ under the condition that $J_{\alpha\theta}(f_k)$ is true; the value of the term exists if all the terms $t_1, ..., t_m$ and all the formulas $f_1, \ldots, f_m$ have values for the interpretation function α and the substitution θ and also there is the only k such that $J_{\alpha\theta}(f_k)$ is true;

5. numerical constants r; $J_{\alpha\theta}(r)$ has the value of the number corresponding to the numerical constant r; $J_{\alpha\theta}(r)$ does not depend on α and θ;

6. R and also $J_{\alpha\theta}(R)$ is the set of all the real numbers; $J_{\alpha\theta}(R)$ does not depend on α and θ;

6. $t_1 + t_2$, or $t_1 - t_2$, or $t_1 * t_2$, or t_1 / t_2 where t_1 and t_2 are terms; $J_{\alpha\theta}(\ t_1 + t_2) = J_{\alpha\theta}(\ t_1) + J_{\alpha\theta}(t_2)$, i.e. $J_{\alpha\theta}(t_1 + t_2)$ is the sum of $J_{\alpha\theta}(t_1)$ and $J_{\alpha\theta}(t_2)$; the value of the term exists if both $J_{\alpha\theta}(t_1)$ and $J_{\alpha\theta}(t_2)$ are numbers; $J_{\alpha\theta}(\tau)$ where τ is $t_1 - t_2$ or $t_1 * t_2$ or t_1 / t_2 is defined in such a way;

11. $t_1 \uparrow t_2$, where t_1 and t_2 are terms; $J_{\alpha\theta}(t_1 \uparrow t_2) = J_{\alpha\theta}(t_1) \uparrow J_{\alpha\theta}(t_2)$, i.e. $J_{\alpha\theta}(t_1 \uparrow t_2)$ is $J_{\alpha\theta}(t_2)$-th power of the number $J_{\alpha\theta}(t_1)$; the value of the term exists if both $J_{\alpha\theta}(t_1)$ and $J_{\alpha\theta}(t_2)$ are numbers and $J_{\alpha\theta}(t_2)$-th power of the number $J_{\alpha\theta}(t_1)$ exists;

12. $\varnothing$ and also $J_{\alpha\theta}(\varnothing)$ is the empty set; $J_{\alpha\theta}(\varnothing)$ does not depend on α and θ;

13.$\{t_1, ..., t_k\}$, where $t_1, ..., t_k$ are terms; $J_{\alpha,\theta}(\{t_1, ..., t_k\}) = \{J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)\}$, i.e. the value of the term is the set those elements are $J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_k)$; the value of the term exists if for the interpretation function α and the substitution θ the terms $t_1, ..., t_k$ have values;

14.{ }t, where t is a term; $J_{\alpha\theta}(\{\ \}t)$ is the set of all the finite subsets (including the empty set and maybe $J_{\alpha\theta}(t)$) of the set $J_{\alpha\theta}(t)$; the value of the term exists if $J_{\alpha\theta}(t)$ is a set;

15. a quantifier construction $\{(v_1: t_1)\ldots(v_m: t_m)\ f\}$ (intensionality quantifier); $J_{\alpha\theta}(\{(v_1: t_1)\ldots(v_m: t_m)\ f\})$ is the subset of the set of admissible substitutions for $(v_1: t_1)\ldots(v_m: t_m)$ that $J_{\alpha\theta}(f)$= true; the value of the term exists if the formula f has a value for all the admissible substitutions;

16. a quantifier construction $\{(v_1: t_1)\ldots(v_m: t_m)\ t\}$ (quantifier of set transformation); $J_{\alpha\theta}(\{(v_1: t_1)\ldots(v_m: t_m)\ t\})$ is the set of all the values of the term $J_{\alpha\theta}(t)$, where θ belongs to the set of admissible substitutions for $(v_1: t_1)\ldots(v_m: t_m)$; the value of the term exists if $J_{\alpha\theta}(t)$ is a set and if for the interpretation function α and for any admissible substitution θ the term t has a value;

17. $t_1 \cup t_2$, or $t_1 \cap t_2$, or $t_1 \setminus t_2$, where t_1 and t_2 are terms; $J_{\alpha\theta}(t_1 \cup t_2) = J_{\alpha\theta}(t_1) \cup J_{\alpha\theta}(t_2)$, i.e. $J_{\alpha\theta}(t_1 \cup t_2)$ is the union of the sets $J_{\alpha\theta}(t_1)$ and $J_{\alpha\theta}(t_2)$; the value of the term exists if both $J_{\alpha\theta}(t_1)$ and $J_{\alpha\theta}(t_2)$ are sets; $J_{\alpha\theta}(\tau)$ where τ is $t_1 \cap t_2$ or $t_1 \setminus t_2$ is defined in such a way;

18. $\mu(t)$, where t is a term; $J_{\alpha\theta}(\mu(t))$ is the cardinality of the set $J_{\alpha\theta}(t)$; the value of the term exists if $J_{\alpha\theta}(t)$ is a finite set;

19. $t_1 \Uparrow t_2$, where t_1 and t_2 are terms; $J_{\alpha\theta}(t_1 \Uparrow t_2) = J_{\alpha\theta}(t_1) \Uparrow J_{\alpha\theta}(t_2)$, i.e. $J_{\alpha\theta}(t_1 \Uparrow t_2)$ is the set $J_{\alpha\theta}(t_1)$ raised to the Cartesian power $J_{\alpha\theta}(t_2)$; the value of the term exists if $J_{\alpha\theta}(t_1)$ is a set and $J_{\alpha\theta}(t_2)$ is a positive integer; the operation "$\Uparrow$" has all the properties of the Cartesian power but associativity: $J_{\alpha\theta}((t_1 \Uparrow t_2) \Uparrow t_3) \neq J_{\alpha\theta}(t_1 \Uparrow (t_2 * t_3))$;

20. $<t_1, ..., t_m>$, where $t_1, ..., t_m$ are terms; $J_{\alpha\theta}(<t_1, ..., t_m>) = <J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_m)>$, i.e. the value of the term is the m-tuple composed of the values $J_{\alpha\theta}(t_1), ..., J_{\alpha\theta}(t_m)$; the value of the term exists if for the interpretation function α and the substitution θ the terms $t_1, ..., t_m$ have values;

21. $\pi(t_1, t_2)$, where t_1 and t_2 are terms; $J_{\alpha\theta}(\pi(t_1, t_2)) = \pi(J_{\alpha\theta}(t_1), J_{\alpha\theta}(t_2))$, i.e. the value of the term is the $J_{\alpha\theta}(t_1)$-th projection of the tuple (of an element of a Cartesian product) $J_{\alpha\theta}(t_2)$; the value of the term exists if $J_{\alpha\theta}(t_2)$ is a m-tuple and $J_{\alpha\theta}(t_1)$ is a positive integer not greater than m;

22. length(t), where t is a term; $J_{\alpha\theta}(\text{length}(t))$ is the number of elements in the tuple $J_{\alpha\theta}(t)$; the value of the term exists if $J_{\alpha\theta}(t)$ is a tuple.

The formulas are:

1. $t_1 = t_2$, or $t_1 \neq t_2$, where t_1 and t_2 are terms; $J_{\alpha\theta}(t_1 = t_2) \Leftrightarrow J_{\alpha\theta}(t_1) = J_{\alpha\theta}(t_2)$, i.e. the value of the formula is true if and only if the values $J_{\alpha\theta}(t_1)$ and $J_{\alpha\theta}(t_2)$ are the same; the formula has a value if for the interpretation function α and the substitution θ the both terms t_1 and t_2 have values; $J_{\alpha\theta}(t_1 \neq t_2)$ is defined in such a way;

2. $t_1 > t_2$, or $t_1 < t_2$, or $t_1 \leq t_2$, or $t_1 \geq t_2$, where t_1 and t_2 are terms; $J_{\alpha\theta}(t_1 > t_2) \Leftrightarrow J_{\alpha\theta}(t_1) > J_{\alpha\theta}(t_2)$, i.e. the value of the formula is true if and only if the value $J_{\alpha\theta}(t_1)$ is greater than the value $J_{\alpha\theta}(t_2)$; the formula has a value if both $J_{\alpha\theta}(t_1)$ and $J_{\alpha\theta}(t_2)$ are numbers; $J_{\alpha\theta}(\phi)$ where ϕ is $t_1 < t_2$, or $t_1 \leq t_2$, or $t_1 \geq t_2$ is defined in such a way;

3. $t_1 \in t_2$, or $t_1 \notin t_2$, where t_1 and t_2 are terms; $J_{\alpha\theta}(t_1 \in t_2) \Leftrightarrow J_{\alpha\theta}(t_1) \in J_{\alpha\theta}(t_2)$, i.e. the value of the formula is true if and only if the value $J_{\alpha\theta}(t_1)$ belongs to the set $J_{\alpha\theta}(t_2)$; the formula has a value if for the interpretation function α and the substitution θ the term t_1 has a value and $J_{\alpha\theta}(t_2)$ is a set; $J_{\alpha\theta}(t_1 \notin t_2)$ is defined in such a way;

4. $t_1 \subset t_2$, or $t_1 \subseteq t_2$, or $t_1 \not\subset t_2$, where t_1 and t_2 are terms; $J_{\alpha\theta}(t_1 \subset t_2) \Leftrightarrow J_{\alpha\theta}(t_1) \subset J_{\alpha\theta}(t_2)$, i.e. the value of the formula is true if and only if the set $J_{\alpha\theta}(t_1)$ is a proper subset of the set $J_{\alpha\theta}(t_2)$; the formula has a value if both $J_{\alpha\theta}(t_1)$ and $J_{\alpha\theta}(t_2)$ are sets; $J_{\alpha\theta}(\phi)$ where ϕ is $t_1 \subseteq t_2$, or $t_1 \not\subset t_2$ is defined in such a way.

Conclusions

In this article the kernel of the extendable language of applied logic has been introduced. Any applied logic theory is characterized by a set (perhaps empty) consisting of the standard extension and specialized extensions of the language. The article also defines the standard extension of the language. The standard extension introduces syntactic constructions for some special languages of mathematical logic and also arithmetic and set-theoretic constants, operations and relations.

References

[Guarino, 1998] Guarino N. Formal Ontology and Information Systems. In Proceeding of International Conference on Formal Ontology in Information Systems (FOIS'98), N. Guarino (ed.), Trento, Italy, June 6-8, 1998. Amsterdam, IOS Press.

[Studer et al, 1998] Studer R., Benjamins V.R., Fensel D. Knowledge Engineering: Principles and Methods. In Data & Knolwedge Engineering, 1998, 25, p. 161-197.

[Wielinga et al, 1994] Wielinga, B., Schreiber A.T., Jansweijer W., Anjewierden A. and van Harmelen F. Framework and Formalism for Expressing Ontologies (version 1). ESPRIT Project 8145 KACTUS, Free University of Amsterdam deliverable, DO1b.1, 1994.

[van Heijst et al, 1996] van Heijst G., Schreiber A.Th., Wielinga B.J. Using Explicit Ontologies in KBS Development. In Intern. Jornnal of Human and Computer Studies, 1996, 46 (2-3), pp. 183-292.

[Artemjeva et al, 1995] Artemjeva I.L., Gavrilova T.L., Kleshchev A.S. Domain Models with Elementary Objects. In Scientific-Technical Information, Series 2, 1995, № 12. P. 8-18 (in Russian).

[Artemjeva et al, 1996] Artemjeva I.L., Gavrilova T.L., Kleshchev A.S. Logical Relationship Systems with Elementary Objects. In Scientific-Technical Information, Series 2, 1996, № 1, pp. 11-18 (in Russian).

[Artemjeva et al, 1997a] Artemjeva I.L., Gavrilova T.L., Kleshchev A.S. Logical Domain Models of the Second Order. In Scientific-Technical Information, Series 2, 1997, № 6, pp. 14-30 (in Russian).

[Artemjeva et al, 1997b] Artemjeva I.L., Gavrilova T.L., Kleshchev A.S. Logical Relationship Systems with Parameters. In Scientific-Technical Information, Series 2, 1997, № 7, pp. 19-23 (in Russian).

[Kleshchev et al, 1998] Kleshchev A.S., Artemjeva I.L., Gavrilova T.L., Surov V.V. Application of Logical Relationship Systems for Expert System Development. In Appl. of Advanced Information Technologies: Proc. of the Forth World Congress on Expert Systems, 16-20 March 1998, Mexico City Cognisant Communication Corporation, 1998, vol.1: 500-510.

Authors' Information

Alexander S. Kleshchev – kleschev@iacp.dvo.ru

Irene L. Artemjeva – artemeva@iacp.dvo.ru

Institute for Automation & Control Processes, Far Eastern Branch of the Russian Academy of Sciences

5 Radio Street, Vladivostok, Russia

INFORMATION PROCESSING IN A COGNITIVE MODEL OF NLP

Velina Slavova, Alona Soschen, Luke Immes

Abstract: A model of the cognitive process of natural language processing has been developed using the formalism of generalized nets. Following this stage-simulating model, the treatment of information inevitably includes phases, which require joint operations in two knowledge spaces – language and semantics. In order to examine and formalize the relations between the language and the semantic levels of treatment, the language is presented as an information system, conceived on the bases of human cognitive resources, semantic primitives, semantic operators and language rules and data. This approach is applied for modeling a specific grammatical rule – the secondary predication in Russian. Grammatical rules of the language space are expressed as operators in the semantic space. Examples from the linguistics domain are treated and several conclusions for the semantics of the modeled rule are made. The results of applying the information system approach to the language turn up to be consistent with the stages of treatment modeled with the generalized net.

Keywords: Cognitive model, Natural Language Processing, Generalized Net, Language Information System

ACM Classification Keywords: I.2.7 Natural Language Processing;

Introduction

Natural language processing (NLP) is a complex cognitive function, representing a complicated subject for modeling and formal description. The trial in this wok is to elaborate a reliable formal model of NLP in order to propose a tool for analyzing the process and to examine the possibilities for further implementations.

The formal model of NLP, presented here, is elaborated using a cognitive science approach. The intention is to take into consideration as much as possible the essential cognitive principles that most cognitive scientists agree with:

1). The mental system has a limited capacity - the amount of information that can be processed by the system is constrained.

2) A control mechanism is required to oversee the encoding, transformation, processing, storage, retrieval and utilization of information.

3) The constructing of meaning is a dynamic process resulting of a two-way flow of information – the flow, gathered through the senses (Bottom-up processing) and the flow of the information, which is stored and classified in the memory (Top-down processing)[1].

4) The human organism has been genetically prepared to process and organize information in specific ways. In the further description we'll consider that all these functions are performed within a system, called *"cognitive system".*

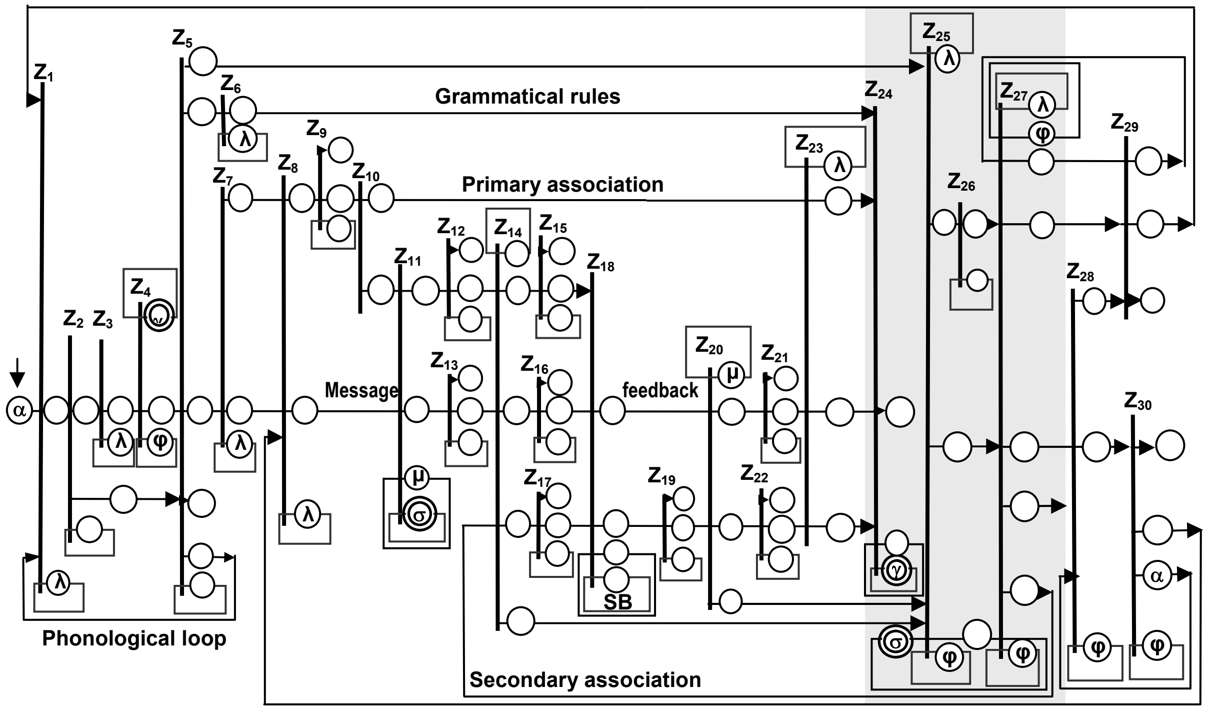


Figure 1. AGN - Generalized Net model of process of message acquisition

A formal model of NLP has been elaborated, using the mathematical formalism of Generalized Nets (GN). The obtained Net, called AGN (Figure 1) gives a formal description of the cognitive process of treatment of a language message, arriving on the input of the auditory system (Bottom), processed stage by stage and conduced

[1] Concerning the cognitive aspects of language processing, the constraints on linguistic performance come mostly from the top-down information processing.

to the mind (Top) through several parallel pathways. AGN[1] corresponds to the cognitive system, providing a control mechanism for overseeing the transformation, processing, storage and retrieval of information.

AGN treats language message, consisting in sentence-fragments, formally expressed as a sequence of α-tokens. Each α-token, travelling from the input to the final position of the net, is submitted to consequent treatments, performed on the transitions Z_i. AGN transitions Z_1 – Z_{29} imitate the phases of the process of speech perception. The information, obtained by the system, is formalized as α-token's characteristics, which are acquired when crossing the transitions. The net al.ows modeling the interaction between the Bottom-up and Top-down information flows. The Top-down information flow assigns new characteristics to the "travelling up" signal. To imitate the parallel "emergence" of the gathered information of different type, the α-token splits (see for example Z_5), follows different pathways and terminates by fusing all obtained characteristics into an internal lexical and semantic representation of the message content.

The Top-knowledge, stored in Long Term Memory (LTM) is organized in two related spaces[2] – the language space (as a system of lexical units and rules) and the semantic space (the semantic representation of the world as a system of semantic primitives and rules). They have respectively two underlying structures - the word-forms graph WG, expressed by the γ-token, and the semantic net NSet, expressed by the σ-token. The result of the Top-down flow is stored in WG as "*expectation*" of word-forms[3]. Four sources of expectation are modeled. Two are related to the language - the memorized language practice, called "*primary association*" and the knowledge of *grammatical rules*. Two other sources of expectation are due to semantic activation: the listening-comprehension *message feedback* (caused directly by the word-forms in the message) and the "*secondary association*" (semantic activation, accumulated because of the sequence of the message word-forms). Tokens γ and σ are thought as structures, which elements accumulate expectation/activation. AGN has access to each of them on two places - one 'retrieval' place (Z_4 and Z_{11}) and one place for storing expectation/activation (Z_{24} and Z_{25})

The knowledge of language is represented by a number of λ-tokens, each skilled with a treatment procedure as characteristic. They are on transitions: Z_1 - with: "Segmentation procedure *Seg*"; Z_3 - with: "Phonemes recognition procedure *Rec*"; Z_6 - with: "Grammatical features and dependencies procedure *Gr*"; Z_7 - with: "Lexeme representative retrieval procedure *InL*"; Z_8 - with: "Primary association procedure *Ass1*"; Z_{23} - with: "Lexeme members retrieval procedure *Lexemize*"; Z_{25} - with: "word-forms concordance procedure *TreeBranches*"; and Z_{27} - with: "Syntax structure discovery procedure *Parse*". The cognitive processes are expressed by φ-tokens, and the "mental dictionary" (relations on WG x NSet) - by μ-tokens. All this tokens, with procedures as characteristics, are 'turning' over the corresponding transitions during the time-period of AGN functioning.

Initially, token α enters the net with characteristics "Phonological features". At transition $Z_{1,}$ a λ–token "Language knowledge – prosody" skilled with the Segmentation procedure transforms the input to a "Sentence segmented into word-form segments", given to the α-token as characteristics. Transition Z_2 simulates auditory sensory memory. Transition Z_3 corresponds to the stage of phoneme recognition. Transition Z_4 corresponds to the comparison of the recognized phonetic content with the lexical knowledge retrieved from WG. Transition Z_5 accepts or rejects the retrieved word-form for further treatment (AGN is supposed to identify the input word-forms using the *expectation,* gathered in the units of WG). Transitions Z_6 to Z_{24} represent a Working Memory (WM) Sub-Net[4] where the two information flows meet[5] and generate expectation. Multiple transitions in this part are connected to LTM-tokens, producing lists of LTM knowledge (lexeme representatives, synonyms, homonyms,

[1] AGN has been presented in a series of papers in the domain of information technologies and cognitive modelling in linguistics (see for example [Kujumdjieff, Slavova, 2000]. Here we follow the numbering and the names, accepted in the complete formal description of the net, given in [Slavova, 2004].

[2] Most cognitive researchers agree on the different nature of the language knowledge and the conceptual knowledge, including their separate localization on the cortex.

[3] It is known that the capacities of speech perception do not allow capturing all the pronounced phonemes. In fact, the cognitive system constructs the missed, but 'expected' content of the message. The same top-down phenomenon is available when reading texts.

[4] This sub-net is presented in details in [Slavova, Atanassov, 2004).

[5] According to the most part of the existing in cognitive science theories and models of memory, the Top-down and the Bottom-up flows meet using Working Memory resources.

concepts, attributes etc.) The sub-net imitates limited WM resource and ‘concentration’ on the message content by retaining only the heads of the lists, sorted following the accumulated activation. On Z_{24} the expectation from all pathways is overlapped and stored in the elements of WG (see also figure 2, token γ on place l_{82}). Transition Z_{25} simulates the activation of NSet by the message word-forms. Transition Z_{26} is a working memory for lexical units. Transition Z_{27} expresses the process of analyzing the entire sentence. Transition Z_{28} simulates the extraction of basic semantic roles (with a feedback from the message memory content). Transition Z_{29} simulates rechecking (if the semantic roles can not be properly discovered). The α-token is finally stored with all obtained characteristics in the message memory, represented on transition Z_{30}.

The Generalized Net approach has allowed formalizing, on a high level of abstraction, the cognitive process of message acquisition. This representation allows incorporation of sub-nets and separate modules, such as databases, neuron nets etc. Such an approach starts to be used in hybrid nets in AI [see Atanassov, K., 1998].

The Problem – Parallel Language and Semantic Treatment

A big part of the treatment procedures, introduced on AGN transitions, such as *“segmentation procedure”* or *“expectation dependent retrieval from WG”*, are easy to be imagined. The problem is to conceive the procedures, which run on the transitions after the WM sub-net. The tracking of the cognitive process has lead to base them on semantic and language knowledge at once. The presented work gives further development of AGN by analyzing the procedures, which perform simultaneously in the language and the semantic space.

Transition Z_{25} (figure 2) simulates two processes, which run in parallel. The first is the activation of the semantic space by the message word-forms W and corresponds to building a mental image of W in terms of concepts and features. The second one is the detection of related words in the sentence and occurs at the moment when the grammatical features **and** the semantic image of W are discovered. It is supposed that the cognitive system first assembles a fractional representation of the sentence-meaning structure (coupled words for example) by consulting the semantic net for incompatibilities, as the grammatically determined word-chains have to be coherent with the meaning of the corresponding concepts and/or features. The formal description of Z_{25} is:

$$Z_{25} = < \{l_{12}, l_{49}, l_{69}, l_{70}, l_{83}, l_{89}\}, \{l_{83}, l_{85}, l_{87}, l_{88}, l_{89}\},$$

	l_{83}	l_{85}	l_{87}	l_{88}	l_{89}
l_{12}	*false*	*true*	*true*	*false*	*false*
l_{49}	*false*	*false*	*true*	*true*	*false*
l_{69}	*false*	*false*	*false*	*true*	*false*
l_{70}	*false*	*false*	*true*	*false*	*false*
l_{83}	*true*	*false*	*true*	*false*	*false*
l_{89}	*false*	*false*	*false*	*true*	*true*

$$, \vee(\wedge(l_{70}, l_{83}, l_{12}), \wedge(l_{89}, \vee(l_{49}, l_{69})))>.$$

The α-token enters transition Z_{25} with the following characteristics, coming from:

position l_{12} – W, word-form assumed to be perceived (on transition Z_5) with its grammatical feathers GrFtrs;

position l_{49} - Nt ct (from the message feed back pathway) – the head of the list of semantic net elements NSet, which correspond to W (the correspondence is found on Z_{11} using the mental dictionary - µ-token);

position l_{69}, - NtSBlist - the first n of list of NSet elements, which correspond to the received up to the moment W - s, arranged following the number of their manifestations in a semantic buffer SB (secondary association).

On transition Z_{25}:

λ-token “Language knowledge - syntax and grammar” turns on place l_{83} with:

“word-forms concordance - Procedure *TreeBranches* ”;

σ-token, the Semantic net NSet stays on place l_{70}:

“Semantic net elements – NSet”;

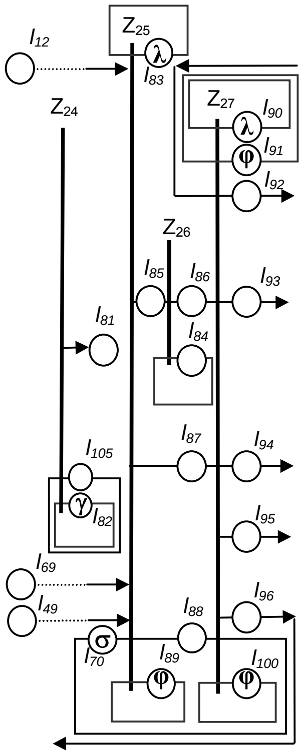


Figure 2. Two of the transitions, based on language and semantics

φ-token “Cognitive process - semantic activation” turns on place l_{89} with

“Storage of activation in NSet nodes - Procedure *SemA*”;

After Z_{25} , on place l_{88} σ-token takes the characteristic:

“ANet = *SemA* (Nt ct, NtSBlist) – activation of NSet elements”

Token α obtains in place l_{87} the characteristic:

“ParSynStr = *TreeBranches* (NSet, GrFtrs) – Partial syntax structure.”

Tokens do not change their characteristics on places l_{83}, l_{85} and l_{89}.

Transition Z_{26} is WM buffer for W, queued on l_{84} and transmitted to l_{86}.

On transition Z_{27}, the following procedures are running:

λ-token “Language syntax knowledge” stays on place l_{90} with:

“Syntax structure discovery - Procedure *Parse*”;

φ-token “comparing semantics and syntax” stays on l_{91} with:

“Comparing – Procedure *Comp*”;

φ-token “focus determination” stays on place l_{100} with:

“Semantic center localization - Procedure *DetSC*”.

Transition Z_{27} expresses the mental process of analyzing the entire sentence after its last word-form has been perceived. It is assumed that two parallel processes take place at this time-moment: the sentence syntax structure is clarified and the semantic focus Nt^1 of the sentence is detected (NSet element, staying higher in NSet’s hierarchy). The brought by α-token information, acquired before entering Z_{27}, consists of: partial syntax representation, word-forms W in the lexical buffer content and activation of the corresponding nodes of NSet. It is supposed that the syntax structure of the sentence is recognized with semantic justification.

Z_{27} = < {l_{86}, l_{87}, l_{88}, l_{90}, l_{91}, l_{99}, l_{100}}, {l_{70}, l_{90}, l_{91}, l_{92}, l_{93}, l_{94}, l_{95}, l_{96}, l_{100}},

	l_{70}	l_{90}	l_{91}	l_{92}	l_{93}	l_{94}	l_{95}	l_{96}	l_{100}
l_{86}	*false*	*false*	*false*	*true*	*true*	*true*	*false*	*false*	*false*
l_{87}	*false*	*false*	*false*	*false*	*true*	*true*	*false*	*false*	*false*
l_{88}	*true*	*false*	*false*	*false*	*true*	*true*	*true*	*true*	*false*
l_{90}	*false*	*true*	*false*	*false*	*true*	*true*	*false*	*false*	*false*
l_{91}	*false*	*false*	*true*	*false*	*true*	*true*	*false*	*false*	*false*
l_{99}	*false*	*false*	*false*	*true*	*true*	*true*	*false*	*false*	*false*
l_{100}	*false*	*false*	*false*	*false*	*false*	*false*	*true*	*true*	*true*

$,\vee(\wedge(\vee(l_{86}, l_{99}), \wedge(l_{88}, l_{90}, l_{91})), \wedge(l_{88}, l_{100}))$ >

In places l_{93} and l_{94} the α-token obtains the characteristic:

“TRes = *Comp* (*Parse*(Buff, ParSynStr), NSet) - Obtaining complete syntax structure”

and in places l_{95} and l_{96} the α-token obtains the characteristic:

“Nt^1 = *DetSC* (NSet, ANet) – momentary semantic center”

Tokens do not change their characteristics on places l_{90}, l_{91}, l_{92}, l_{70} and l_{100}. The procedure *Comp* may have two results: 1. TCF (Tree Construction Failed); 2. TREE+ WsC (Syntax tree with W corrected - WsC).

The procedure *Comp* (*Parse*(Buff, ParSynStr), NSet) on Z_{27} is applied on the retained in a STM buffer for W and on the result of *TreeBranches* (NSet, GrFtrs), running on $Z_{25.}$ Both procedures are based on NSet and on the grammatical features of W (discovered on Z_4).

The Language as an Information System

The generalized net model suggests that we have to formalize Z_{25} and Z_{27} through procedures, running by using in parallel semantic and language knowledge and related to the elements and to the rules in the two knowledge spaces. A sequence of questions appears concerning the simultaneous operations in the two spaces. In AGN, the correspondences between them are given by the 'mental dictionary', containing the relations between the lexical units in WG and the elements of the semantic net NSet, but the structures of the two spaces are independent. Are there rules that allow joint operations in the two spaces? Are they established on principles for mapping the structures of the two spaces? That needs to examine the human language in a general way.

Let us present human language as an Information System (IS) – a Language Information System (LIS). One of the primary goals of a human language is to assure the information exchange between individuals. Information, residing as *internal cognitive representation* of the individual H_1 is first presented as language-coded information, communicated to another individual H_2, and interpreted to *internal cognitive representation* of individual H_2. It is intriguing to provide the example of one home-made sign language, created and utilized by two deaf sisters. The used pointing gestures were found to be part of lexical terms referring to present and non-present objects, persons and places. Some gestures occupied fixed positions in sentences, apparently used as grammatical terms. Oral movements were frequently used together with manual signs, and their functions may be classified as lexical, adverbial, and grammatical (Torigoe et al., 2002). This strongly suggests the existence of the *innate* mechanism for mental representations (Hauser 2002, Chomsky 2004).

Imagine we have to construct a LIS. Let us apply the used in the technology domain procedure. For conceiving an IS, the representation "*input - treatment block – output"* is used. On its input, an IS receives *data* and *resources* and on its output - obtains *informative products*. The treatment block runs on the bases of a particular *model and method for data-processing,* which includes a number of *rules* and *operators* on data. So:

1. The *resources* of LIS are the human cognitive resources – *static* (long term memory and working memory), and *operational* (operators that human mind performs on the available operable substances).

2. Data. The mind-operable content of the data-source (figure 3) has to be transmitted as data, operable in language. The functioning of the data-source has to be presented as a model, a system of elements with determined roles, reproducing how the cognitive system operates when performing its tasks. Let us call the components on this model "semantic primitives". A structure of data-containers has to be assembled in the language and matched to the structure of the semantic primitives. Data-values have to be accorded to all distinct entities, available and operable in the source, and stored in the corresponding data-containers. This approach is well-known in the IS domain (see for example Codd, 79).

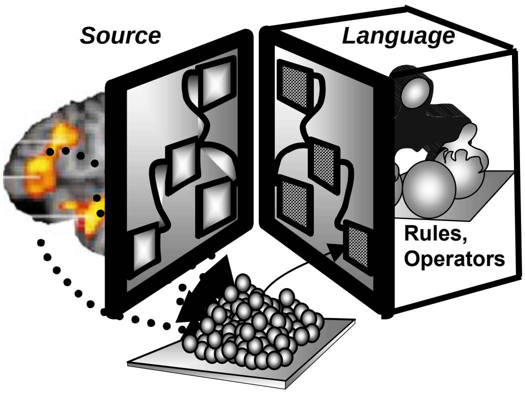


Figure 3. The language, seen as information system

3. The *treatment block* requires conceiving a set of *rules* and *operators* on data. This set has to allow generating larger units, reproducing the processing in the source. The cognitive system will operate data following these rules, so they must be expressible by means of the operators, which run on the semantic primitives.

The *final product* of the language has to create an accurate internal representation of individual H_2, who receives the LIS output. The 'decoding' is done with the active participation of H_2 (the 'Top-down" information processing), which presupposes that H_2 knows the language **and** possess a semantic description of the world.

The information system reasoning shows that the accurate functioning of a LIS strongly relies on the internal semantic representation. The language is constructed on the bases of the semantic primitives and the mind operators on them. The claim is that in all languages must exist interactions between the purely language features and their semantic fundamentals. This could give a basis for the joint semantic-language operations, leading to a solution of the problem concerning the procedures on Z_{25} and Z_{27}. The next step is to examine the relationships between the grammatical level and the semantic one, starting from concrete working examples.

Cognitively Based LIS on the Example of Russian

We assume that there is a common general underlying semantic scheme for all languages. Then it will follow that any grammatical rule can be represented as consisting of some semantic primitives as internal representations, which are mind-operable. We did a trial to show how LIS operates on the example of a concrete syntactic representation in one specific language. Secondary predication is a grammatical particularity in Russian, allowing variability of case marking (Instrumental/Nominative) on secondary predicates. Example:

(1) a. Maria prišla ustalaja-nom.
b. Maria prišla ustaloj-instr.
Mary arrived tired.

In Russian, we can make the notion of arriving tired, simultaneously being contrasted with some non-tired state, using the instrumental case. In the nominative case, we only know that 'she arrived tired', with no past reference to any other possible state. The semantics of Russian secondary predication has been examined a lot by the specialists in linguistics and the obtained results and explanations are not uniform.

We have followed the LIS reasoning and we constructed a database (DB) in which exists simultaneously the language level and the semantic level, with their structures, interconnected.

Examples of statements, taken from the linguistics studies of secondary predication, (53 sentences) wore stored in the table Examples. The cases in Russian are used as markers for the grammatical annotation of the examples. Data, expressing the language and semantics spaces, have been organized in tables as follows (figure 4):

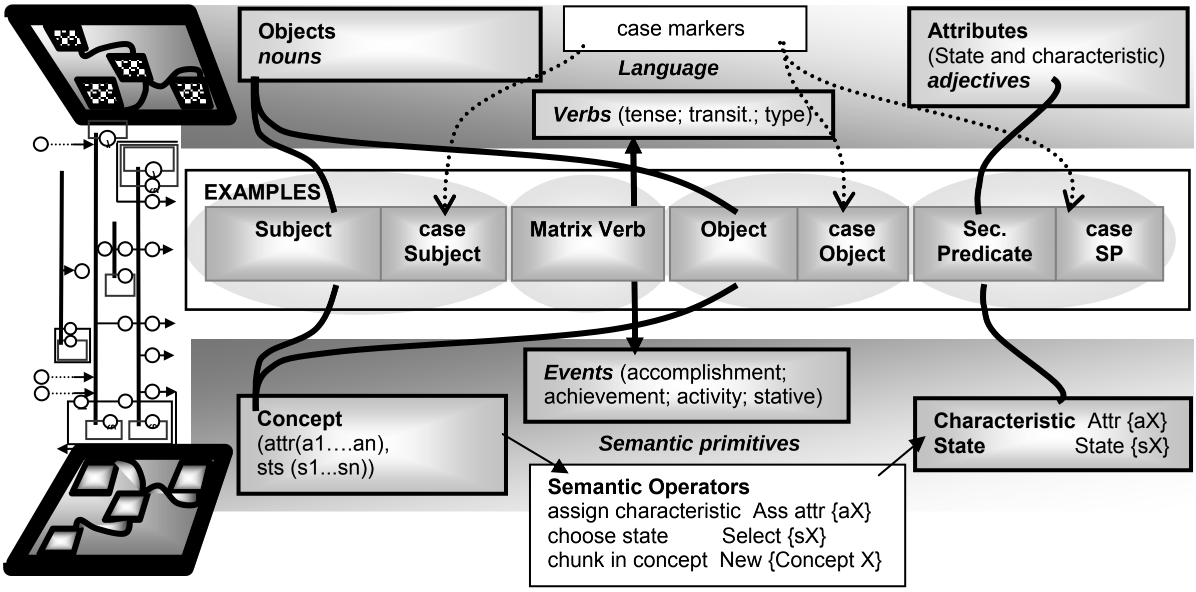


Figure 4. Language – semantics database design

Table Objects stores all *concepts* from the examples, with their names in different languages. The *attributes* - characteristics or possible states, are stored in a separate table. The verbs with their grammatical features are stored in the table 'Verbs' (they are introduced in the field "Matrix Verb" of the table Examples as foreign key). The *events*, providing the underlying semantic features of the verbs, are expressed as attributes of the verbs (foreign key). Events are stored separately with their semantic features following some of the existing classifications. With this construction of the DB, the grammatical rules of case marking can be examined independently of the events' structure. For the purposes of this analysis, we employ a revised version of the 'event structure' (Davidson 1967, Vendler1957, Verkuyl 2001), examining the language semantics primitives.

The statements are assigned two levels of representation - phases of semantic-levels-translation: The first step accords to a lexical item its basic semantic category. The categories that we used are *concept*, *characteristic*, *state* and *event.* We assume that the grammatical level, expressed by means of case markers, implies running of semantic operators. For our examples we took as basic operators the following set: "*assign characteristic*: Ass attr {aX}", "*choose state:* Select {sX}" and "*chunk in concept*: New {Concept X}".

The second representation of the statements gives the result of applying the semantic operator. Using queries over the modeled in the DB parameters, we checked several guesses about the semantic interpretation, coming from the linguistics domain. For example, as the running of queries over the events characteristics does not give any indication of changing states, so the conclusion is that the meaning of the matrix verb events is not influenced by the case marking. It is interesting to see the result of queries, which put together (taking data from the corresponding tables) all language labels (Russian and English in Table 1), the semantic markers and the verb-event information. Here are a few of the examples for transitive verbs (*Re* - the results of the first phase of semantic-levels-translation, *Se* – the semantics of the sentence, obtained after the second phase):

Table 1

	13a.			
Ex	**Ja/-nom**	**Pokupaju**	**banany/-acc**	**spely/-instr**
	I/-nom	*Buy*	*bananas/-acc*	*ripe/-instr*
Re	**Ja (concept)**	**Pokupaju**	**banany (concept)**	**spely (state)**
	I (concept)	*Buy*	*bananas (concept)*	*ripe (state)*
	attr(a1….an), sts (s1….sn)		attr(a1….an), sts (s1….sn)	
Se	**Ja (concept)**	**Pokupaju**	**banany-spely (selected state)**	
	I (concept)	*Buy*	*bananas-ripe (selected state)*	
	attr(a1….an), sts (s1….sn)		in state {sX}	
	Activity	(<…,<sn,sn+k,k=1>,…>)	'I buy bananas ripe.'	
	16a1			
Ex	**Don/-nom**	**Pišet**	**pis'mo/-acc**	**ustal/-instr**
	Don/-nom	*Writes*	*letter/-acc*	*tired/-instr*
Re	**Don (concept)**	**Pišet**	**pis'mo (concept)**	**ustal (state)**
	Don (concept)	*Writes*	*letter (concept)*	*tired (state)*
	attr(a1….an), sts (s1….sn)		attr(a1….an), sts (s1….sn)	state {sX}
Se	**Don-ustal (selected state)**	**Pišet**	**pis'mo (concept)**	
	Don-tired (selected state)	*Writes*	*letter (concept)*	
	in state {sX}		attr(a1….an), sts (s1….sn)	
	Activity	(<…,<sn,sn+k,k=1>,…>)	'Don writes letter tired.'	

It come out that the use of a canonical underlying semantic scheme of events, objects, states and attributes explains the semantics of the grammatical rule of secondary predication in a clear way:

The case marking of the secondary predicate implies meaning of a "*choose state:* Select {sX}" operator in the case of instrumental and an "*assign characteristic*: Ass attr {aX}" operator in the case of nominative. The grammatical rule of secondary predication is applied to concepts and plays a role of a "choice of state" operator without influencing the structure of the matrix verb's event. The semantic-level interpretation of all statements shows that the event structure plays its role for the meaning of the sentences in an independent way.

This clear representation of secondary predication may be implemented in several ways. We constructed a Neuron Net (figure 5), which performs the treatment of secondary predication grammatical rule, as the aim is to include further the treatment of other rules.

Our task is to model the more explicit Russian syntax. So, we choose our essential cognitive features, with just a sufficient enough neural network. The nodes of our neural network are conventional symbols: nouns, adjectives, adverbs or verbs. Activation of the node occurs when a sufficient threshold value is reached. The sum is used for AND operation, and 1 is use for OR operation. Initially, all inputs are OFF. There is no learning component.

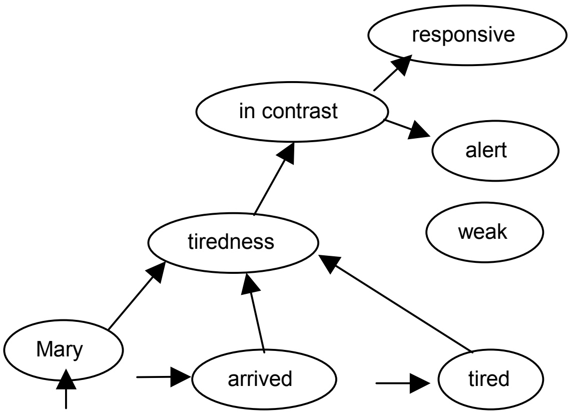


Figure 5. Neuron Net

Word order is not part of the model, but left up to the actual language. Because Russian has a more flexible word order, and the net meaning is the same for both Russian and English (as concerns ordering of words), then assume words have been entered in the right order.

The neuron net imitates: Instr. case marking on *tired* in *Mary arrived tired* triggers a state of *tiredness.* Being in a state of tiredness gives rise to an explicit state *in contrast*, then non-tired states: of: *alert*, and *responsive*.

The advantages of this representation are 1) semantics and syntax are combined; 2) knowledge engineering is much easier, including maintenance, because symbols are used, instead of dynamic numeric values. Clearly, the neuron net has to have a much richer set of semantics for a practical system.

Conclusion

The supposal that the cognitive system treats in parallel the semantic and the language knowledge space was made on the bases of two formal representations: the AGN model of the cognitive process and the representation of the language as an information system. The results of these two formal approaches are in agreement. The 'semantic' database representation of primary/secondary predication on the example of Russian was used in the analysis of the links between basic semantic units and grammar. Grammatical rules of the language space are expressible as operators in the semantic space. Some important linguistics conclusions wore made on this base.

It is interesting to analyze the content of Table 1. It became clear that for intransitive verbs the *choice of state* operator is applied always to the state of subject, but for transitive verbs it can also be applied to the state of object. Obviously in the statement 13a the state 'ripe' can not be accorded to the subject 'I' and in 16a1 the state 'tired' is not for the object 'letter'. But these two statements are absolutely correct, they are not ambiguous and in use in both languages. From the point of view of AGN treatment, on Z_{25} the procedure *TreeBranches* has to 'attach' the word-form "ripe" to "bananas" and the word-form "tired" to "I", as it consults the semantic space NSet, where the concepts and their attributes are known. The further development of this work necessitates formalizing in a detailed way the AGN-s part Z_{25} Z_{27}, associated to LIS on order to implement this part of the model.

Bibliography

[Atanassov, 1998] K. Atanassov. Generalized Nets in Artificial Intelligence. Vol. 1: Generalized nets and Expert Systems. "Prof. M. Drinov" Academic Publishing House, Sofia

[Becker, Immes, 1987] L.A. Becker, L. Immes. An approach to automating knowledge acquisition for Expert Systems: annotated traces -> Diagnostic Hierarchies. In: ACM Conference on Computer Science, p133-137.

[Chomsky, 2004] N. Chomsky. The Generative Enterprise Revisited. Mouton de Gruyter.

[Codd, 1979] E. F. Codd. Extending the Database Relational Model to Capture more Meaning. In: ACM TODS, Vol 4, n 4.

[Davidson, 1967] D. Davidson. The Logical Form of Action Sentences. In: N. Rescher, ed., The Logic of Decision and Action, Pittsburgh University Press, Pittsburgh.

[Hauser et al., 2002] M. Hauser, N. Chomsky and W.T. Fitch. The Faculty of Language: What is it, who has it, and how did it evolve? In: Science Vol. 298.

[Kujumdjieff, Slavova, 2000] T. Kujumdjieff, V. Slavova. An association-based model of natural language comprehension. In: International Journal "Information theories & applications", Vol. 7 Number 4, 169-174.

[Torigoe et al., 2002] Torigoe, Takashi, Takei, Wataru. Descriptive Analysis of Pointing and Oral Movements in a Home Sign System. In: Sign Language Studies 2.3, Spring 2002;

[Slavova, 2004] V. Slavova. A generalized net for natural language comprehension. In: Advanced Studies in Contemporary Mathematics, vol 8, Ku-Duk Press, 131-153.

[Slavova, Atanassov,2004] V. Slavova, K. Atanassov. A generalized net for working memory and language. In: Text Processing and Cognitive Technologies, VII International conference «Cognitive modeling in linguistics», Varna, 90-101;

[Soschen, 2003] A. Soschen. On Subject and Predicates in Russian. Ph.d. Dissertation, Univ. Ottawa.

[Vendler, Zeno.,1957] Vendler, Zeno. Verbs Verbs and Times. In: Philosophical Review 66, 143-160.

[Verkuyl, H.J., 2001] H.J. Verkuyl. Aspectual Composition: Surveying the Ingredients. In: Proceedings of the Utrecht Perspectives on Aspect Conference December 2001, OTS.

Author's Information

Velina Slavova – New Bulgarian University, Department of Computer Science, 21 Montevideo str., 1618 Sofia, Bulgaria, e-mail: vslavova@nbu.bg

Alona Soschen – Massachusetts Institute of Technology, MIT Department of Linguistics and Philosophy 77 Massachusetts Ave. 32-D808Cambridge, MA 02139-4307, USA, e-mail: soschen@mit.edu

Luke Immes – Multisegment Co., 20 Williamsburg Court, suite 14 Shrewsbury, MA 01545, USA, e-mail: li@multisegment.com

MATHEMATICAL MODEL OF RE-STRUCTURING COMPLEX TECHNICAL AND ECONOMIC STRUCTURES

May Kornijchuk, Inna Sovtus, Eugeny Tsaregradskyy

Abstract: Research and development of mathematical model of optimum distribution of resources (basically financial) for maintenance of the new (raised) quality (reliability) of complex system concerning, which the decision on its re-structuring is accepted, is stated. The final model gives answers (algorithm of calculation) to questions: how many elements of system to allocate on modernization, which elements, up to what level of depth modernization of each of allocated is necessary, and optimum answers are by criterion of minimization of financial charges.

Keywords: system, re-structuring, quality, reliability.

ACM Classification Keywords: I.6.3 Simulation and Modeling: Applications

Introduction

By development of new complex systems, and increase of their efficiency while in service the important factor of increase of adequacy and reliability of mathematical models an estimation of a level of their reliability is ability of the description, formalization and the account in these models of an opportunity of management of reliability [1]. The increment of reliability *u* due to rational management of reliability is achieved by perfection of algorithm of a system's mode of operation variations, a variation of actions on technical and to preventive maintenance, because that reduction of failure rate after rational procedure of procedural works depends on a level of optimization of this procedure. Reduction of intensity of a refusal's stream, change of its probable structure of limited after action can be achieved also by special modes of external influences. So, for example, separate kinds of integrated circuits at a radioactive irradiation sharply raise accuracy of a presence of parameters in necessary borders [1]. However the time of their life essentially decreases. Realization of such procedure when the system carries out the important and responsible task nevertheless can be quite justified. Value of such task allows neglecting reduction of general time of life of an element due to strict preservation of parameters in the certain limits, though in a smaller time interval. In such situations there is a problem about management of reliability with the purpose of optimization of system by the certain criterion.

Let's investigate such variant of complex system, which is characterized by the linear block diagram of its process of functioning. It can be complex economic structure, or complex technical system, or such system which has both economic and technical properties, that is complex technical-economic structure. Let such complex system contains *n* subsystems, are connected functional consistently. The system has the certain degree of quality of functioning. We for convenience shall interpret this level its level of reliability *P*. At the certain stage of operation of system (for example, by virtue of development of market relations of the country) there comes such moment when the condition of efficiency of system's process of functioning at level *P* becomes insufficient. There is a problem of increase of reliability up to level $P^* = P + + \Delta P$. At this stage of ability to live of many systems there are problems on their reorganization and re-structuring. Thus the powerful part of the general investments should be allocated for an innovation.

Active investments in innovational process of an element can result in an essential gain of reliability of an element or probability of stay of parameters in the necessary limits [2, 3]. This most reliability of system as a whole rises. To increase it is possible would be as much as you want if it was not connected to material inputs. Expansion of quantity of the modernized elements and their parameters, results increase of depth of innovational process of each element in the appropriate growth of charges for realization of all re-structuring. Objectively there is some point (value of a level of completion) after which to increase charges for re-structuring from practical reasons begins inexpedient. There is a task on search of such point, namely: where it is necessary to stop at a choice of a level of depth of completion of a separate element. For all system the problem is, what quantity of elements to subject modernization that economic feasibility of carried out work was not lost. And further, when the question for system as a whole about quantity of the elements chosen for modernization is solved, there will be a following problem: which among *n* elements need to be chosen, and up to which level of depth each of the chosen elements needs to be finished? Clearly, that there is a sense to solve these problems only in view of expenses for completion. Thus the level of completion of an element from the point of view of reliability of system will be determined by a new necessary level of reliability of this element in comparison with reliability *p* which is incorporated during manufacture of an element. Depth of completion, modernization of *i* an element of system quantitatively can be to characterize a gain x_i reliability of this element due to innovations. Thus if *i* the element or unit is not finished it (is not modernized), it is logical to put value $x_i = 0$. Then depth of completion or reconstruction of all system can be characterized *a n*-dimensional vector $\vec{x} = \{x_1, x_2, ..., x_n\}$, where x_i, $i = \overline{1,n}$ - essence the designations entered above. If to designate through p_i reliability of *i* system's an element prior to the beginning of completion after reconstruction reliability of an element will be equal $p_i + x_i$. Reliability of system before realization of modernization was the function $P = P(\vec{p}),$ after modernization it will increase and will get new value:

$$P = P(\vec{p}, \vec{x}).$$

Mathematical Model

It is necessary to investigate dependence of charges for completion of an element and its quality acquired due to completion. That is to investigate dependence of charges and an additional gain of probability x_i, taking into account thus an increment of reliability due to management. Further we shall name this dependence the function of cost of an element's completion [4] and we shall designate it through *K*(*x*). As well as at the decision of any other practical problem, here we can not take advantage of real (empirical) dependence of charges for completion and a gain of reliability. For use of this dependence it is necessary to formalize it, construct analytical function which full enough approximates the given empirical dependence that is it is necessary to construct mathematical model of dependence. We shall take the mathematical model of dependence which is investigated and constructed in work [4], and it is proved by economic analogues [5], namely:

$$K(x) = \frac{Ax}{q - u - x},$$

in which q = 1 - p, and the factor A is defined from experimental data for concrete system. As we consider complex system with consecutive connection of subsystems (elements) we suppose, that the system is not a reliable (there comes refusal of system) if even one of n parameters (elements) of system has left for allowable limits. In the assumption of consecutive connection of elements and the account of an increment of reliability due to modernization and managements the equation of reliability of system will have the form:

$$P = \prod_{i=1}^{n} (p_i + u_i + x_i). \tag{1}$$

The total function of charges for modernization of system is equal

$$K(\vec{x}) = \sum_{i=1}^{n} \frac{A_i x_i}{q_i - u_i - x_i}. \tag{2}$$

Let's notice, that function of charges for the modernization (2) and the equation of reliability (1) make sense only with the certain restrictions which are consequence of their physical contents, namely:

$$1.\ K(x) \geq 0; \quad 2.\ K(0) = 0; \quad 3.\ \lim_{x \uparrow 1-p} K(x) = \infty. \tag{3}$$
$$x_i \geq 0, \quad x_i < q_i - u_i, \quad i = \overline{1,n}.$$

Now the problem was reduced to a task of nonlinear mathematical programming. More precisely the problem consists in a finding of such decision of the equation (1) which would provide under conditions (3) minimum of function (2). We already marked, that similar tasks of the theory of nonlinear programming still demand researches. Therefore we are very much limited in a choice of methods for a finding of decisions. For definition of optimum requirements to completion of each element, that is for a finding of a conditional minimum of function (2) in view of restriction on variables (1) and boundary conditions (3) we shall take advantage of the modified [6] method of Lagrange's uncertain multipliers.

Application of the modified method is necessary because direct use of Lagrange's method is impossible because the area of restrictions (3) is open. Therefore we use one substantial equality - restriction, namely (1). This equation information absorbs the others.

We build the function similar to Lagrange's function, we have:

$$f(\vec{x}, \lambda) = \sum_{i=1}^{n} \frac{A_i x_i}{q_i - u_i - x_i} + \lambda \left[\prod_{i=1}^{n} \left(p_i + u_i + x_i \right) - P^* \right]. \tag{4}$$

From construction of function (4) it is visible, that in it restriction as the equation (1) is used only and boundary conditions as inequalities are not taken into account. Such step, generally speaking, is not absolutely true, but in this case it is justified by information capacity of restriction (1). It will be proved below, that offered use of restriction (1) results in obligatory performance of boundary conditions (3), and procedure of search of the decision provides it. At the same time the offered way releases the decision of a problem in the chosen method from analytical bulkiness of introduction of huge quantity of unknown additional multipliers such as Lagrange's multipliers.

For definition of components of a vector $\vec{x}$, for an optimum set x_i, i = 1, 2, ..., n we shall construct system from n the equations, having calculated individual derivatives from Lagrange's function $f(\vec{x}, \lambda)$ on all variable x_i and having equated them to zero. We shall receive:

$$\frac{\partial f}{\partial x_i} = \frac{A_i q_i}{\left(q_i - u_i - x_i \right)^2} - \frac{\lambda}{p_i + u_i + x_i} \prod_{j=1}^{n} \left(p_j + u_j + x_j \right) = 0, \tag{5}$$
$$i = 1, 2, \ldots, n.$$

The system of the equations (5) contains (n+1) unknown: n unknown x_i and the unknown λ. If to this system to add the equation (1) we shall receive full system n +1 the equations with n +1 unknown. The received system of the equations (5), (1) has the decision, and besides the only thing that will be proved below.

For convenience we shall transform the equations (5) to a kind:

$$-\lambda\prod_{j=1}^{n}\left(p_j+u_j+x_j\right)=\frac{A_iq_i\left(p_i+u_i+x_i\right)}{\left(q_i-u_i-x_i\right)^2},\qquad(6)$$
$$i=1,\ 2,\ ...,\ n.$$

We have allocated to the left the identical parts of the equations (5). As expression which is in the left part of each equation (6) is a constant in relation to an index *i* also the right parts of the equations should coincide. We admit from some physical or practical conditions it is possible to come to a conclusion, that for *k* an element of system completion is necessary. It means $x_k > 0$. We shall take *k* the equation of system (6) and instead of expression in the left part we shall substitute equal to it *i* expression from the current equation of system. Then the system of the equations (6) will be copied as follows:

$$\frac{A_iq_i\left(p_i+u_i+x_i\right)}{\left(q_i-u_i-x_i\right)^2}=\frac{A_kq_k\left(p_k+u_k+x_k\right)}{\left(q_k-u_k-x_k\right)^2},\qquad(7)$$
$$i=1,\ 2,\ ...,\ k-1,\ k+1,\ ...\ n.$$

As we have fixed a choice *k* an element expression on the right has function which does not depend from *i*, therefore we shall designate the common right part of parity (7) through B_k, namely:

$$B_k=\frac{A_kq_k\left(p_k+u_k+x_k\right)}{\left(q_k-u_k-x_k\right)^2}\qquad(8)$$

Further we pass from system (7) to a classical kind of the square-law equation:

$$B_k\left(x_i+u_i-q_i\right)^2-A_iq_i\left(x_i+u_i-q_i\right)-A_iq_i=0,\qquad(9)$$
$$i=1,\ 2,\ ...,\ k-1,\ k+1,\ ...\ n.$$

The system of the equations of the second degree (9) is received. It is equivalent to system of the equations (5). We research it on existence and uniqueness of the decision. With this purpose we shall enter functions

$$\varphi_i(x_i)=A_iq_i(p_i+u_i+x_i),$$
$$\psi_i(x_i)=B_k(x_i+u_i-q_i)^2,$$
$$i=1,\ 2,\ ...,\ k-1,\ k+1,\ ...\ n.$$

These are functions according to the right and left parts of the equation $\varphi(x_i) = \psi(x_i)$ which is the system equivalent (9).

As for *i* the equations of system at value $x_i = q_i - u_i$ functions get values $\varphi_i(q_i-u_i)=A_iq_i>0$ and $\psi_i(q_i- u_i) = 0$ the parity takes place $\varphi_i(q_i-u_i)>\psi_i(q_i-u_i)$. Graphically it means, that the straight line φ_i (x_i) will cross a parabola ψ_i (x_i) in two points x_i. One of the points x_i is located more to the left $q_i - u_i$, that is $x_i < q_i - u_i$. Other point is located to the right of $q_i - u_i$, that is $x_i > q_i - u_i$. Thus, the second roots of system of the equations (9) have no sense. They do not represent for us interest as they lie outside of area of physically legal values x_i, outside of area of existence of the decision of a task in view and do not satisfy to boundary conditions (3). From properties of monotony and a continuity of functions φ_i (x_i) and ψ_i (x_i) follows, that the point of crossing of these curves more to the left of value $x_i = q_i - u_i$ is unique. Then the unique left roots of system of quadratics (9) look like:

$$x_i=q_i-u_i+\frac{1-\sqrt{1+2\alpha_i}}{\alpha_i},\qquad(10)$$
$$i=1,\ 2,\ ...,\ k-1,\ k+1,\ ...,\ n,$$

in which $\alpha_i=2B_k\,/\,A_iq_i$.

Let's calculate values of the entered functions in the left ends of an interval [0; $q_i - u_i$) - ranges of definition of variables x_i as in the right ends we have already compared boundary their values. At $x_i = 0$ function $\varphi_i(x_i)$ accepts value $\varphi_i(0)=A_iq_i\left(p_i+u_i\right)$, and function $\psi_i(x_i)$ turns to number $\psi_i(0)=B_k\left(q_i-u_i\right)^2$.

For those *i* elements, for which $\varphi_i(0) > \psi_i(0)$, the point of crossing of curves $\varphi_i(x_i)$ and $\psi_i(x_i)$ lays more to the left of zero. Values of a root turn out $x_i < 0$. It contradicts a physical nature of a variable x_i and means what to modernize such element it is not necessary. It is necessary to put $x_i = 0$, thus the condition (3) will be provided. If for determined i an element of value of the entered functions will be razed to the ground $\varphi_i(0) = \psi_i(0)$, that is $\frac{A_i q_i(p_i + u_i)}{(q_i - u_i)^2} = B_k$, that the point of crossing of diagrams $\varphi_i(x_i)$ and $\psi_i(x_i)$ will get on an axis of ordinates. Thus the root $x_i = 0$ will turn out. It also means what to modernize an element it is not necessary. Thus the boundary condition (3) was provided automatically, as proves the statement stated above.

As a result we have: on the chosen value of a level of a gain of reliability x_k for k an element of system unequivocally we determine a level of completion i (i = 1, 2, ..., k – 1, k + 1, ..., n) an element x_i by means of parities:

$$x_i = \begin{cases} 0, & \frac{A_i q_i(p_i + u_i)}{(q_i - u_i)^2} \geq B_k, \\ q_i - u_i + \frac{1 - \sqrt{1 + 2\alpha_i}}{\alpha_i}, & \frac{A_i q_i(p_i + u_i)}{(q_i - u_i)^2} < B_k, \end{cases} \quad i = 1, 2, ..., k-1, k+1, ..., n. \tag{11}$$

Constructed mathematical model (11) solves a task in view of distribution of resources (financial) for optimum re-structuring complex system with the purpose of increase of its quality (reliability) in view of management of reliability from level P up to level $P^* = P + \Delta P$.

Conclusion

Using the initial information on system (reliability, cost of elements and the units, the available unsatisfactory level of reliability P new P^* required degree of quality of system in which achievement consists sense of re-structuring), mathematical model (11) is constructed. It gives answers to all questions of achievement of concrete decisions which are formulated in statement of a problem and are textually formulated in the summary in the beginning of work. We shall note simplicity of calculations, applied and their transparent interpretation at practical use of the developed model.

Bibliography

1. *Severtsev N.A.* Reliability of complex systems in operation and improvement. (*Северцев Н.А.* Надежность сложных систем в эксплуатации и отработке.) – М.: Высшая школа, 1989. – 432 с.
2. *Valter Y.* Stochastic models in economy. (*Валтер Я.* Стохастические модели в экономике.) – М.: Статистика, 1976. – 232 с.
3. *Sarkisjan S.A., Kaspin V.I., Lisichkin* V.A. Theory of forecasting and acceptance of decisions. (*Саркисян С.А., Каспин В.И., Лисичкин В.А.* Теория прогнозирования и принятия решений.) – М.: Высшая школа, 1977. – 352 с.
4. *Kornijchuk M.T.* Mathematical of model of optimization and estimation reliability and efficiency of functioning complex RTS. (*Корнийчук М.Т.* Математические модели оптимизации и оценивания надежности и эффективности функционирования сложных РТС.) – К.: КВИРТУ ПВО, 1980.–280 с.
5. *Meskon N.H., Albert M., Hedorne F.* Bases of management. (*Мескон Н.Х., Альберт М., Хедоурн Ф.* Основы менеджмента.) Пер. с англ. – М.: «Дело», 1992. – 702 с.
6. *Kornijchuk M.T., Sovtus I.K.* Stochastic models of information technologies of optimization of reliability of complex systems. (*Корнійчук М.Т., Совтус І.К.* Стохастичні моделі інформаційних технологій оптимізації надійності складних систем.) – К.: КВІУЗ, 2000. – 316 с.

Authors' Information

May Kornijchuk – doctor of sciences, professor. Kiev National Economic University, Prosp. Pobeda, 54/1, Kiev-03680, Ukraine, e-mail: sovtus@bigmir.net

Inna Sovtus – doctor of sciences, professor. Kiev National Economic University, Prosp. Pobeda, 54/1, Kiev-03680, Ukraine, e-mail: sovtus@bigmir.net

Eugeny Tsaregradsky – Kiev National Economic University, Prosp. Pobeda, 54/1, Kiev-03680, Ukraine, e-mail: YTsaregradskyy@bmw.ua

DIAGARA: AN INCREMENTAL ALGORITHM FOR INFERRING IMPLICATIVE RULES FROM EXAMPLES

Xenia Naidenova

Abstract: An approach is proposed for inferring implicative logical rules from examples. The concept of a good diagnostic test for a given set of positive examples lies in the basis of this approach. The process of inferring good diagnostic tests is considered as a process of inductive common sense reasoning. The incremental approach to learning algorithms is implemented in an algorithm DIAGaRa for inferring implicative rules from examples.

Keywords: Incremental and non-incremental learning, learning from examples, machine learning, common sense reasoning, inductive inference, good diagnostic test, lattice theory.

ACM Classification Keywords: I.2.6 Artificial Intelligence: Learning; K.2.3. Concept Learning

Introduction

Our approach to machine learning problems is based on the concept of a good diagnostic (classification) test. This concept has been advanced firstly in the framework of inferring functional and implicative dependencies from relations [Naidenova and Polegaeva, 1986]. But later the fact has been revealed that the task of inferring all good diagnostic tests for a given set of positive and negative examples can be formulated as the search of the best approximation of a given classification on a given set of examples and that it is this task that all well known machine learning problems can be reduced to [Naidenova, 1996].

We have chosen the lattice theory as a model for inferring good diagnostic tests from examples from the very beginning of our work in this direction. We believe that it is the lattice theory that must be the mathematical theory of common sense reasoning. One can come to this conclusion by analyzing both the fundamental work in the psychological theory of intelligence [Piaget, 1959], and the experience of modeling thinking processes in the framework of artificial intelligence. The process of objects' classification has been considered in [Shreider, 1974] as an algebraic idempotent semi group with the unit element. An algebraic model of classification and pattern recognition based on the lattice theory has been advanced in [Boldyrev, 1974]. A lot of experience has been obtained on the application of algebraic lattices in machine learning: the works of Finn and his disciples [Finn, 1984], [Kuznetsov, 1993], the model of conceptual knowledge of Wille [1992], the works of the French group [Ganascia, 1989]. The following works are devoted to the application of algebraic lattices for extracting classifications, functional dependencies and implications from data: [Demetrovics and Vu, 1993], [Mannila and Räihä, 1992], [Mannila and Räihä, 1994], [Huntala, et al., 1999], [Cosmadakis, et al., 1986], [Naidenova and Polegaeva, 1986], [Megretskaya, 1989], [Naidenova, et al., 1995a], [Naidenova, et al., 1995b], and [Naidenova, 1992].

An advantage of the algebraic lattices approach is based on the fact that an algebraic lattice can be defined both as an algebraic structure that is declarative and as a system of dual operations with the use of which the elements of this lattice can be generated. This approach allows us to investigate the processes of inferring good classification tests as inductive reasoning processes. In the following part of this chapter, we shall describe our decomposition of the inductive inferring process into subtasks and operations that conform to the operations and subtasks of the natural human reasoning process.

This paper is organized as follows. The concept of a good diagnostic test is introduced and the problem of inferring all good diagnostic tests for a given classification on a given set of examples is formulated. The next section contains the description of a mathematical model underlying algorithms of learning reasoning. We propose a decomposition of learning algorithms into operations and subtasks that are in accordance with human reasoning operations. In the second part of this paper, the concepts of an essential value and an essential example are also introduced and an incremental learning algorithm DIAGaRa is described. The paper ends with a brief summary section.

The Concept of a Good Classification Test

Our approach for inferring implicative rules from examples is based on the concept of a good classification test. A good classification test can be understood as an approximation of a given classification on a given set of examples [Naidenova, 1996]. On the other hand, the process of inferring good tests realizes one of the known canons of induction formulated by J. S. Mill, namely, the joint method of similarity-distinction [Mill, 1900].

A good diagnostic test for a given set of examples is defined as follows. Let *R* be a table of examples and *S* be the set of indices of examples belonging to *R*. Let *R*(*k*) and *S*(*k*) be the set of examples and the set of indices of examples from a given class *k,* respectively.

Denote by *FM* = *R*/*R*(*k*) the examples of the classes different from class *k*. Let *U* be the set of attributes and *T* be the set of attributes values (values, for short) each of which appears at least in one of the examples of *R*. Let *n* be the number of examples of *R*. We denote the domain of values for an attribute *Atr* by *dom*(*Atr*), where $Atr \in U$.

By *s*(*a*), $a \in T$, we denote the subset $\{i \in S$: '*a*' appears in t_i, $t_i \in R\}$, where $S = \{1, 2, .., n\}$.

Following [Cosmadakis, et al., 1986], we call *s*(*a*) the interpretation of $a \in T$ in *R*. It is possible to say that *s*(*a*) is the set of indices of all the examples in *R* which are covered by the value *a*.

Since for all *a*, $b \in dom(Atr)$, $a \neq b$ implies that the intersection $s(a) \cap s(b)$ is empty, the interpretation of any attribute in *R* is a partition of *S* into a family of mutually disjoint blocks. By *P*(*Atr*), we denote the partition of *S* induced by the values of an attribute *Atr*. The definition of *s*(*a*) can be extended to the definition of *s*(*t*) for any collection *t* of values as follows: for *t*, $t \subseteq T$, if $t = a_1\, a_2 \ldots a_m$, then $s(t) = s(a_1) \cap s(a_2) \cap \ldots \cap s(a_m)$.

Definition 1. A collection $t \subseteq T$ ($s(t) \neq \varnothing$) of values, is a diagnostic test for the set *R*(*k*) of examples if and only if the following condition is satisfied: $t \not\subset t^*$, $\forall\, t^*$, $t^* \in FM$ (the equivalent condition is $s(t) \subseteq S(k)$).

To say that a collection *t* of values is a diagnostic test for the set *R*(*k*) is equivalent to say that it does not cover any example belonging to the classes different from *k*. At the same time, the condition $s(t) \subseteq S(k)$ implies that the following implicative dependency is true: 'if *t*, then *k*.

It is clear that the set of all diagnostic tests for a given set *R*(*k*) of examples (call it '*DT*(*k*)') is the set of all the collections *t* of values for which the condition $s(t) \subseteq S(k)$ is true. For any pair of diagnostic tests t_i, t_j from *DT*(*k*), only one of the following relations is true: $s(t_i) \subseteq s(t_j)$, $s(t_i) \supseteq s(t_j)$, $s(t_i) \approx s(t_j)$, where the last relation means that $s(t_i)$ and $s(t_j)$ are incomparable, i.e. $s(t_i) \not\subset s(t_j)$ and $s(t_j) \not\subset s(t_i)$. This consideration leads to the concept of a good diagnostic test.

Definition 2. A collection $t \subseteq T$ ($s(t) \neq \varnothing$) of values is a good test for the set *R*(*k*) of examples if and only if the following condition is satisfied: $s(t) \subseteq S(k)$ and simultaneously the condition $s(t) \subset s(t^*) \subseteq S(k)$ is not satisfied for any t^*, $t^* \subseteq T$, such that $t^* \neq t$.

Good diagnostic tests possess the greatest generalization power and give a possibility to obtain the smallest number of implicative rules for describing examples of a given class *k*.

The Characterization of Classification Tests

Any collection of values can be irredundant, redundant or maximally redundant.

Definition 3. A collection *t* of values is irredundant if the following condition is satisfied: $(\forall v)$, $(v \in t)$, $s(t) \subset s(t/v)$.

If a collection *t* of values is a good test for *R*(*k*) and, simultaneously, it is an irredundant collection of values, then any proper subset of *t* is not a test for *R*(*k*).

Definition 4. Let $X \to v$ be an implicative dependency which is satisfied in *R* between a collection $X \subseteq T$ of values and the value *v*, $v \in T$. Suppose that a collection $t \subseteq T$ of values contains *X*. Then the collection *t* is said to be redundant if it contains also the value *v*.

If *t* contains the left and the right sides of some implicative dependency $X \to v$, then the following condition is satisfied: $s(t) = s(t/v)$. In other words, a redundant collection *t* and the collection *t*/*v* of values cover the same set of examples.

If a good test for *R*(*k*) is a redundant collection of values, then some values can be deleted from it and thus obtain an equivalent good test with a smaller number of values.

Definition 5. A collection $t \subseteq T$ of values is maximally redundant if for any implicative dependency $X \to v$, which is satisfied in *R*, the fact that *t* contains *X* implies that *t* also contains *v*.

If *t* is a maximally redundant collection of values, then for any value $v \notin t$, $v \in T$ the following condition is satisfied: $s(t) \supset s(t \cup v)$. In other words, a maximally redundant collection *t* of values covers the number of examples greater than the collection $(t \cup v)$ of values.

Any example *t* in *R* is a maximally redundant collection of values because for any value $v \notin t$, $v \in T$ $s(t \cup v)$ is equal to $\varnothing$.

If a diagnostic test for a given set *R*(*k*) of examples is a good one and it is a maximally redundant collection of values, then by adding to it any value not belonging to it we get a collection of values which is not a good test for *R*(*k*).

Table - 1. Example 1 of Data Classification. (This example is adopted from [Ganascia, 1989]).

Index of Example	Height	Color of Hair	Color of Eyes	Class
1	Short	Blond	Blue	1
2	Short	Brown	Blue	2
3	Tall	Brown	Embrown	2
4	Tall	Blond	Embrown	2
5	Tall	Brown	Blue	2
6	Short	Blond	Embrown	2
7	Tall	Red	Blue	1
8	Tall	Blond	Blue	1

For example, in Table 1 the collection '*Blond Blue*' is a good irredundant test for class 1 and simultaneously it is maximally redundant collection of values. The collection '*Blond Embrown*' is a test for class 2 but it is not good test and simultaneously it is maximally redundant collection of values.

The collection '*Embrown*' is a good irredundant test for class 2. The collection '*Red*' is a good irredundant test and the collection '*Tall Red Blue*' is a maximally redundant and good test for class 1.

It is clear that the best tests for pattern recognition problems must be good irredundant tests. These tests allow construction of the shortest implicative rules with the highest degree of generalization.

An Approach for Constructing Good Irredundant Tests

Let *R*, *S*, *S*(+), *T*, *s*(*t*), $t \subseteq T$, $s \subseteq S$ be as defined earlier. We give the following propositions the proof of which can be found in [Naidenova, 1999].

PROPOSITION 1.

The intersection of maximally redundant collections of values is a maximally redundant collection.

PROPOSITION 2.

Every collection of values is contained in one and only one maximally redundant collection with the same interpretation.

PROPOSITION 3.

A good maximal redundant test for R(k) either belongs to the set R(k) or it is equal to the intersection of q examples from R(k) for some q, $2 \le q \le nt$, where nt is the number of examples in R(k).

One of the possible ways for searching for good irredundant tests for a given class of examples is the following: first, find all good maximally redundant tests; second, for each good maximally redundant test, find all good irredundant tests contained in it. This is a convenient strategy as each good irredundant test belongs to one and only one good maximally redundant test with the same interpretation.

It should be more convenient in the following considerations to denote the set *R*(*k*) as *R*(+) (the set of positive examples) and the set *R*/*R*(*k*) as *R*(-) (the set of negative examples). We will also denote the set *S*(*k*) as *S*(+).

The following Algorithm 1 solves the task of inferring all good maximally redundant tests for a given set of positive examples. The idea of this algorithm has been advanced in [Naidenova and Polegaeva, 1991].

By $s_q = (i_1, i_2, \ldots, i_q)$, we denote a subset of S, containing q indices from S. Let S(test-q) be the set of elements $s = \{i_1, i_2, ..., i_q\}$, $q = 1,2, ..., nt$, satisfying the condition that $t(s)$ is a test for $R(+)$. Here nt denotes the number of positive examples.

We will use an inductive rule for constructing $\{i_1, i_2, ..., i_{q+1}\}$ from $\{i_1, i_2, ..., i_q\}$, $q = 1, 2, ..., nt-1$. This rule relies on the following consideration: if the set $\{i_1, i_2, ..., i_{q+1}\}$ corresponds to a test for $R(+)$, then all its proper subsets must correspond to tests too and, consequently, they must be in S(test-q). Thus the set $\{i_1, i_2, ..., i_{q+1}\}$ can be constructed if and only if S(test-q) contains all its proper subsets. Having constructed the set $s_{q+1} = \{i_1, i_2, ..., i_{q+1}\}$, we have to determine whether it corresponds to the test or not. If $t(s_{q+1})$ is not a test, then s_{q+1} is deleted, otherwise s_{q+1} is inserted in S(test-(q+1)). The algorithm is over when it is impossible to construct any element for S(test-(q+1)).

We use in Algorithm 1 the function to_be_test(t): if $s(t) \cap S(+) = s(t)$ ($s(t) \subseteq S(+)$) then *true* else *false*.

We introduce the mapping $t(s)$ = {intersection of all t_i: $t_i \subseteq T$, $i \in s$}.

Algorithm 1. Inferring all Good Maximally Redundant Tests (GMRTs) for a set $R(+)$ of positive examples.

```
    1. Input: $q = 1$, $R$, $S$, $R(+)$, $S(+)= \{1,2,\ldots, nt\}$, $S$(test-$q$) = $\{\{1\}, \{2\}, ..., \{nt\}\}$.
    Output: the set $TGOOD$ of all GMRTs for $R(+)$.
    2. $S_q$ ::= $S$(test-$q$);
    3. While $||S_q|| \geq q + 1$ do
    3.1 Generating $S(q + 1) = \{s =\{i_1, ..., i_{(q + 1)}\}: (\forall j) (1 \leq j \leq q + 1) (i_1, ..., i_{(j-1)}, i_{(j + 1)}, ..., i_{(q + 1)}\} \in S_q\}$;
    3.2 Generating $S$(test-($q$ + 1)) = $\{s = \{i_1, ..., i_{(q + 1)}\}$: ($s \in S(q + 1)$) & (to_be_test($t(s)$) = $true$)$\}$;
    3.3 $S$(test-$q$) ::= $\{s = \{i_1, ..., i_q\}$: ($s \in S$(test-$q$)) & (($\forall s'$)($s' \in S$(test-($q$ + 1)) $s \not\subset s'$)$\}$;
    3.4. $q$ ::= $q$ + 1;
    3.5. $max$ ::= $q$;
    end while
    4. $TGOOD$::= $\emptyset$;
    5. While $q \leq max$ do $TGOOD$::= $TGOOD \cup \{t(s)$: $s = \{i_1, ..., i_s\} \in S$(test-$q$) $\}$;
    5.1 $q$::= $q$ + 1;
    end while
    end
```

An illustration of inferring GMRTs for the examples of class 2 (see, please, Table 1) is given in Table 2.

The set S_q, $q = 2$ consists of 10 elements {{2,3}, {2,4}, {2,5}, {2,6}, {3,4}, {3,5}, {3,6}, {4,5}, {4,6}, {5,6}}. But t({2,4}), t({2,6}), t({4,5}), and t({5,6}) are not tests for class2, hence we can construct only two elements of the next level for $q = 3$: S_3 = S(test-3) = {{2,3,5}, {3,4,6}}.

As a result, the tests obtained correspond to the following implicative rules: “if COLOR of HAIR = *Brown*, then Class = 2” and “if COLOR of EYES = *Embrown*, then Class = 2”.

Algorithm 1 is also used for inferring all good irredundant tests (GIRTs) contained in a good maximally redundant test.

Now let $t = \{a_1, a_2,\ldots, a_m\} \subseteq T$ be a collection of values that is a GMRT for $R(+)$.

We will use a rule of inductive transition from an element $t_q = (a_1, a_2, \ldots, a_q)$ to another element $t_{q+1} = (a_1, a_2, \ldots, a_{q+1})$, $t_q, t_{q+1} \subseteq T$. But now we are interested in obtaining irredundant collections of values. If $t_{q+1} = (a_1, a_2, \ldots, a_{q+1})$ is irredundant, then all its proper subsets must be irredundant too.

Table - 2. Example of inferring logical rules for Class 2 (Table 1) with the use of Algorithm 1.

S(test-1)	$t(s)$, $s \in S$(test-1)	S(test-2)	$t(s)$, $s \in S$(test-2)	S(test-3)	$t(s)$, $s \in S$(test-3)
{2}	‘*Short Brown Blue*’	{2,3}	‘*Brown*’	{2,3,5}	‘*Brown*’
{3}	‘*Tall Brown Embrown*’	{2,5}	‘*Brown Blue*’		

{4}	'*Tall Blond Embrown*'	{3,4}	'*Tall Embrown*'	{3,4,6}	'*Embrown*'
{5}	'*Tall Brown Blue*'	{3,5}	'*Tall Brown*'		
{6}	'*Short Blond Embrown*'	{3,6}	'*Embrown*'		
		{4,6}	'*Blond Embrown*'		

Having constructed the set $t_{q+1} = (a_1, a_2, \ldots, a_{q+1})$, we have to determine whether it is an irredundant collection of values or not. If t_{q+1} is redundant, then it is deleted, if t_{q+1} is a test, then t_{q+1} is inserted in the set *TGOOD* of all good irredundant tests contained in *t*. If t_{q+1} is irredundant but not a test, then it is a candidate for extension.

The following Algorithm 2 solves the task of inferring all GIRTs contained in a maximally redundant test for a given set of positive examples.

We use in Algorithm 2 the function to_be_irredundant(t)::= if for $(\forall a_i)$ $(a_i \in t)$ $s(t) \neq s(t/\, a_i)$ then *true* else *false*.

Algorithm 2. Inferring all GIRTs contained in a given GMRT for *R*(+)**.**

Input: $q = 1$, *R*, *S*, *R*(+), $t = \{a_1, a_2, \ldots, a_m\}$ – a collection of values – a GMRT, *F*(irredundant – *q*) = $\{\{a_1\}, \{a_2\}, \ldots, \{a_m\}\}$ – the family of irredundant subsets of values with *q* equal to 1.

Output: the set *TGOOD* of all the GIRTs for *R*(+) contained in *t*.

```
1. $F_q$::= $F$(irredundant – $q$ );
1.1 Generating $F$(test-$q$ ) =$\{t = \{a_{i1}, \ldots, a_{iq}\}$: $(t \in F_q)$ & (to_be_test($t$) = $true$)$\}$;
1.2 $F_q$ ::= $F_q \setminus F$(test-$q$) ;
2. While $\|F_q\| \geq q + 1$ do
2.1. Generating $F(q + 1)$ =
= $\{t = \{a_{i1}, \ldots, a_{i(q+1)}\}: (\forall j)\ (1 \leq j \leq q + 1)\ (a_{i1}, \ldots, a_{i(j-1)}, a_{i(j+1)}, \ldots, a_{i(q+1)}\} \in F_q\}$;
2.2. Generating $F$(irredundant – $(q+1)$) :
$F$(irredundant – $(q+1)$) ::= $\{t \in F(q + 1)$: to_be_irredundant($t$) = $true$ $\}$;
2.3. $q$ ::= $q + 1$;
2.4. max ::= $q$;
end while
3. $TGOOD$ ::= $\varnothing$;
4.While $q \leq$ max do
4.1. $TGOOD$ ::= $TGOOD \cup F$(test-$q$);
4.2. $q$::= $q + 1$;
end while
end
```

The Duality of Good Diagnostic Tests

In Algorithms 1 and 2, we used (without explicit definition) correspondences of Galois *G* on $S \times T$ and two relations $S \rightarrow T$, $T \rightarrow S$ [Ore, 1944], [Riguet, 1948]. Let $s \subseteq S$, $t \subseteq T$. We define the relations as follows:

$S \rightarrow T$: $t(s)$ = {intersection of all t_i: $t_i \subseteq T$, $i \in s$} and $T \rightarrow S$: $s(t) = \{i: i \in S, t \subseteq t_i\}$.

Extending *s* by an index *j** of some new example leads to receiving a more general feature of examples:

$(s \cup j^*) \supseteq s$ implies $t(s \cup j^*) \subseteq t(s)$.

Extending *t* by a new value '*a'* leads to decreasing the number of examples possessing the general feature '*ta*' in comparison with the number of examples possessing the general feature '*t*':

$(t \cup a) \supseteq t$ implies $s(t \cup a) \subseteq s(t)$.

We introduce the following generalization operations (functions):

generalization_of(t) = $t' = t(s(t))$; generalization_of(s) = $s' = s(t(s))$.

As a result of the generalization of *s*, the sequence of operations $s \rightarrow t(s) \rightarrow s(t(s))$ gives that $s(t(s)) \supseteq s$. This generalization operation gives all the examples possessing the feature $t(s)$.

As a result of the generalization of t, the sequence of operations $t \rightarrow s(t) \rightarrow t(s(t))$ gives that $t(s(t)) \supseteq t$. This generalization operation gives the maximal general feature for examples the indices of which are in $s(t)$.

These generalization operations are not artificially constructed operations. One can perform mentally a lot of such operations during a short period of time. We give some examples of these operations. Suppose that somebody has seen two films (s) with the participation of Gerard Depardieu ($t(s)$). After that, he tries to know all the films with his participation ($s(t(s))$). One can know that Gerard Depardieu acts with Pierre Richard (t) in several films ($s(t)$). After that, he can discover that these films are the films of the same producer Francis Veber $t(s(t))$.

Namely, these generalization operations will be used in the algorithm DIAGaRa.

The Definition of Good Diagnostic Tests as Dual Objects

We implicitly used two generalization operations in all the considerations of diagnostic tests. Now we define a diagnostic test as a dual object, i.e. as a pair (SL, TA), $SL \subseteq S$, $TA \subseteq T$, $SL = s(TA)$ and $TA = t(SL)$.

The task of inferring tests is a dual task. It must be formulated both on the set of all subsets of S, and on the set of all subsets of T.

Definition 6. Let $PM = \{s_1, s_2, \ldots, s_m\}$ be a family of subsets of some set M. Then PM is a Sperner system [Sperner, 1928] if the following condition is satisfied: $s_i \not\subset s_j$ and $s_j \not\subset s_i$, $\forall(i,j)$, $i \neq j$, $i, j = 1, \ldots, m$.

Definition 7. To find all *Good Maximally Redundant Tests* (GMRTs) for a given class $R(k)$ of examples means to construct a family PS of subsets $s_1, s_2, \ldots, s_j, \ldots, s_{np}$ of the set $S(k)$ such that:

1) PS is a Sperner system;

2) Each s_j is a maximal set in the sense that adding to it the index i of example t_i such that $i \notin s_j$, $i \in S$ implies $s(t(s_j \cup i)) \not\subset S(k)$. Putting it in another way, $t(s_j \cup i)$ is not a test for the class k, so there exists such example t^*, $t^* \in R(-)$ that $t(s_j \cup i) \subseteq t^*$.

The set of all GMRTs is determined as follows: $\{t: t(s_j), s_j \in PS, \forall j, j = 1,..., np\}$.

Definition 8. To find all *Good Irredundant Tests* (GIRTs) for a given class $R(k)$ of examples means to find a family PRT of subsets $t_1, t_2,..., t_j,\ldots, t_{nq}$ of the set T such that:

1) $t_j \not\subset t$ $\forall j, j = 1,..., nq$, $\forall t$, $t \in R/R(k)$ and, simultaneously, $\forall t_j$, $j = 1,..., nq$, $s(t_j) \neq \varnothing$ there does not exist such a collection $s^* \neq s(t_j)$, $s^* \subseteq S$ of indices for which the following condition is satisfied $s(t_j) \subset s^* \subseteq S(k)$;

2) PRT is a Sperner system;

3) Each t_j – a minimal set in the sense that removing from it any value $a \in t_j$ implies $s(t_j \text{ without } a) \not\subset S(k)$.

Decomposition of Good Classification Tests Inferring into Subtasks

The Algorithms 1 and 2 find all the GMRTs and GIRTs for a given set of positive examples but the number of tests can be exponentially large. In this case, these algorithms will be not realistic. Now we consider some decompositions of the problem that provide the possibility to restrict the domain of searching, to predict, in some degree, the number of tests, and to choose tests with the use of essential values and/or examples. This decomposition gives an approach to constructing incremental algorithms of inferring all good classification tests for a given set of examples.

We consider two kinds of subtasks (please, see also [Naidenova, 2001]:

for a given set of positive examples

1) Given a positive example t, find all GMRTs contained in t;

2) Given a non-empty collection of values X (maybe only one value) such that it is not a test, find all GMRTs containing X.

Each example contains only some subset of values from T, hence each subtask of the first kind is simpler than the initial one. Each subset X of T appears only in a part of all examples; hence each subtask of the second kind is simpler than the initial one.

Forming the Subtasks

The subtask of the first kind. We introduce the concept of an example's projection proj(*R*)[*t*] of a given positive example *t* on a given set *R*(+) of positive examples. The proj(*R*)[*t*] is the set *Z* = {*z*: (*z* is non-empty intersection of *t* and *t*′) & ($t' \in R(+)$) & (*z* is a test for a given class of positive examples)}.

If the proj(*R*)[*t*] is not empty and contains more than one element, then it is a subtask for inferring all GMRTs that are in *t*. If the projection contains one and only one element equal to *t*, then *t* is a GMRT.

To make the operation of forming a projection perfectly clear we construct the projection of t_2 = '*Short Brown Blue*' on the examples of the second class (Table 1). This projection includes t_2 and the intersections of t_2 with the other positive examples of the second class, i.e. with the examples t_3, t_4, t_5, t_6 (Table 3).

Table - 3. The Intersections of Example t_2 with the Examples of Class 2.

Index of Example	*Height*	*Color of Hair*	*Color of Eyes*	*Test?*
2	*Short*	*Brown*	*Blue*	*Yes*
3		*Brown*		*Yes*
4				*No*
5		*Brown*	*Blue*	*Yes*
6	*Short*			*No*

In order to check whether an element of the projection is a test or not we use the function to_be_test(*t*) in the following form: to_be_test(*t*) = if $s(t) \subseteq S(+)$ then *true* else *false*, where *S*(+) is the set of indices of positive examples, *s*(*t*) is the set of indices of all positive and negative examples containing *t*. If *S*(-) is the set of indices of negative examples, then $S = S(+) \cup S(-)$ and $s(t) = \{i: t \subseteq t_i, i \in S\}$.

Table - 4. The Projection of the Example t_2 on the Examples of Class 2.

Index of Example	Height	Color of Hair	Color of Eyes	Test?
2	Short	Brown	*Blue*	Yes
3		Brown		Yes
5		Brown	Blue	Yes

The intersection $t_2 \cap t_4$ is the empty set. Hence, the row of the projection with the number 4 is empty. The intersection $t_2 \cap t_6$ is not a test for Class 2 because *s*(*Short*) = {1,2,6} $\not\subset$ *S*(+), where *S*(+) is equal to {2,3,4,5,6}.

Finally, we have the projection of t_2 on the examples of the second class in Table 4.

The subtask turns out to be very simple because the intersection of all the rows of the projection is a test for the second class: *t*({2,3,5}) = '*Brown*', *s*(*Brown*) = {2,3,5} $\subseteq$ *S*(+).

The subtask of the second kind. We introduce the concept of an attributive projection proj(*R*)[*a*] of a given value '*a*'on a given set *R*(+) of positive examples.

The projection proj(*R*)[*a*] = {*t*: ($t \in R(+)$) & ('*a*' appears in *t*)}. Another way to define this projection is: proj(*R*)[*a*] = $\{t_i: i \in (s(a) \cap S(+))\}$. If the attributive projection is not empty and contains more than one element, then it is a subtask of inferring all GMRTs containing a given value '*a*'. If '*a*' appears in one and only one example, then '*a*' does not belong to any GMRT different from this example.

Forming the projection of '*a*' makes sense if '*a*' is not a test and the intersection of all positive examples in which '*a*' appears is not a test too, i.e. $s(a) \not\subset S(+)$ and $t' = t(s(a) \cap S(+))$ is also not a test for a given set of positive examples.

Denote the set $\{s(a) \cap S(+)\}$ by splus(*a*). In Table 1, we have:

S(+) = {2,3,4,5,6}, *splus*(*Short*) → {2,6}, *splus*(*Brown*) → {2,3,5}, *splus*(*Blue*) → {2,5}, *splus*(*Tall*) → {3,4,5}, *splus*(*Embrown*) → *{3,4,6}*, and *splus*(*Blond*) → *{4,6}*.

For the value '*Brown*' we have: *s*(*Brown*) = {2,3,5} and *s*(*Brown*) = *splus*(*Brown*), i.e. *s*(*Brown*) $\subseteq$ *S*(+).

*Analogously for the value '*Embrown*' we have:* s*(*Embrown*) = {3,4,6} and* s*(*Embrown*) =* splus*(*Embrown*), i.e.* s*(*Embrown*)* $\subseteq$ S*(+).*

Table - *5*. The Result of Reducing the Projection after Deleting the Values '*Brown*' and '*Embrown*'

Index of Example	Height	Color of Hair	Color of Eyes	Test?
2	Short		Blue	No
3	Tall			No
4	Tall	Blond		No
5	Tall		Blue	No
6	Short	Blond		No

These values are irredundant and simultaneously maximally redundant tests because *t*({2,3,5}) = '*Brown*' and *t*({3,4,6}) = '*Embrown*'. It is clear that these values cannot belong to any test different from them. We delete '*Brown*' and '*Embrown*' from further consideration with the following result as shown in Table 5.

Now none of the remaining rows of the second class is a test because *s*(*Short*, *Blue*) = {1,2}, *s*(*Tall*) = {3,4,5,7,8}, *s*(*Tall*, *Blond*) = {4,8}, *s*(*Tall*, *Blue*) ={5,7,8}, *s*(*Short*, *Blond*) = {1,6} $\not\subset$ *S*(+). The values '*Brown*' and '*Embrown*' exhaust the set of the GMRTs for this class of positive examples.

Reducing the Subtasks

The following theorem gives the foundation for reducing projections both of the first and the second kind. The proof of this theorem can be found in [Naidenova et al., 1995b].

THEOREM 1.

Let A be a value from T, X be a maximally redundant test for a given set R(+) of positive examples and s(A) $\subseteq$ *s(X). Then A does not belong to any maximally redundant good test for R(+) different from X.*

To illustrate the way of reducing projections, we consider another partition of the rows of Table 1 (see, please Part 1 of this paper) into the sets of positive and negative examples as shown in Table 6.

Let *S*(+) be equal to {4,5,6,7,8}. The value '*Red*' is a test for positive examples because *s*(*Red*) = *splus*(*Red*) = {7}. Delete '*Red*' from the projection. The value '*Tall*' is not a test because *s*(*Tall*) = {3,4,5,7,8} and it is not equal to *splus*(*Tall*) = {4,5,7,8}. Also *t*(*splus*(*Tall*)) = '*Tall*' is not a test. The attributive projection of the value '*Tall*' on the set of positive examples is in Table 7.

Table - 6. The Example 2 of a Data Classification.

Index of Example	Height	Color of Hair	Color of Eyes	Class
1	Short	Blond	Blue	1
2	Short	Brown	Blue	1
3	Tall	Brown	Embrown	1
4	Tall	Blond	Embrown	2
5	Tall	Brown	Blue	2
6	Short	Blond	Embrown	2
7	Tall	Red	Blue	2
8	Tall	Blond	Blue	2

Table - 7. The Projection of the Value *'Tall'* on the Set *R(+).*

Index of Example	Height	Color of Hair	Color of Eyes	Test?
4	Tall	Blond	Embrown	Yes
5	Tall	Brown	Blue	Yes
7	Tall		Blue	Yes
8	Tall	Blond	Blue	Yes

In this projection, *splus*(*Blue*) = {5,7,8}, *t*(*splus*(*Blue*)) = '*Tall Blue*', *s*(*Tall Blue*) = {5,7,8} = splus(*Tall Blue*) hence '*Tall Blue*' is a test for the second class. We have also that *splus*(*Brown*) = {5}, but {5} $\subseteq$ {5,7,8} and, consequently, there does not exist any good test which contains simultaneously the values '*Tall*' and '*Brown*'. Delete '*Blue*' and '*Brown*' from the projection as shown in Table 8.

However, now the rows t_5 and t_7 are not tests for the second class and they can be deleted as shown in Table 9. The intersection of the remaining rows of the projection is '*Tall Blond*'. We have that *s*(*Tall Blond*) = {4,8} $\subseteq$ *S*(+) and this collection of values is a test for the second class.

Table - 8. The Projection of the Value *'Tall'* on *R(+)* without the Values *'Blue'* and *'Brown'.*

Index of Example	Height	Color of Hair	Color of Eyes	Test?
4	Tall	Blond	Embrown	Yes
5	Tall			No
7	Tall			No
8	Tall	Blond		Yes

Table - 9. The Projection of the Value *'Tall'* on *R(+)* without the Examples t_5 and t_7.

Index of Example	Height	Color of Hair	Color of Eyes	Test?
4	Tall	Blond	Embrown	Yes
8	Tall	Blond		Yes

As we have found all the tests for the second class containing *'Tall'* we can delete *'Tall'* from the examples of the second class as shown in Table 10.

Table - 10*.* The Result of Deleting the Value *'Tall'* from the Set *R(+).*

Index of Example	Height	Color of Hair	Color of Eyes	Test?	Class
1	Short	Blond	Blue	Yes	1
2	Short	Brown	Blue	Yes	1
3	Tall	Brown	Embrown	Yes	1
4		Blond	Embrown	Yes	2
5		Brown	Blue	No	2
6	Short	Blond	Embrown	Yes	2
7			Blue	No	2
8		Blond	Blue	No	2

Next we can delete the rows t_5, t_7, and t_8. The result is in Table 11.

The intersection of the remaining examples of the second class gives a test '*Blond Embrown*' because

s(*Blond Embrown*) = *splus*(*Blond Embrown*) = {4,6} $\subseteq$ *S*(+).

Table - 11. The Result of Deleting t_5, t_7*,* and t_8 from the Set *R(+).*

Index of Example	Height	Color of Hair	Color of Eyes	Class
1	Short	Blond	Blue	1
2	Short	Brown	Blue	1
3	Tall	Brown	Embrown	1
4		Blond	Embrown	2
6	Short	Blond	Embrown	2

The choice of values or examples for forming a projection requires special consideration.

In contrast to incremental learning, where the problem is considered of how to choose relevant knowledge to be best modified, here we come across the opposite goal to eliminate irrelevant knowledge not to be processed.

Choosing Values and Examples for the Formation of Subtasks

Next, it is shown that it is convenient to choose essential values in an example and essential examples in a projection for the decomposition of the problem of inferring GMRTs into the subtasks of the first or second kind.

An Approach for Searching for Essential Values

Let *t* be a test for positive examples. Construct the set of intersections {*t* ∩ *t*': *t*' ∈ *R*(-)}. It is clear that these intersections are not tests for positive examples. Take one of the intersections with the maximal number of values in it. The values complementing the maximal intersection in *t* is the minimal set of essential values in *t*.

Next we describe the procedure with the use of which a quasi-maximal subset of *t** that does not correspond to a test is obtained.

We begin with the first value a_1 *t**, then we take the next value a_2 of *t** and evaluate the function to_be_test ({a_1, a_2}). If the value of the function is *false*, then we take the next value a_3 of *t** and evaluate the function to_be_test ({a_1, a_2, a_3})). If the value of the function to_be_test ({a_1, a_2}) is *true*, then the value a_2 of *t** is skipped and the function to_be_test ({a_1, a_3})) is evaluated. We continue this process until we achieve the last value of *t**.

Return to Table 6. Exclude the value '*Red*' (we know that '*Red*' is a test for the second class) and find the essential values for the examples t_4, t_5, t_6, t_7, and t_8. The result is in Table 12.

Consider the value '*Embrown*' in t_6: *splus*(*Embrown*) = {4,6}, *t*({4,6}) = '*Blond Embrown*' is a test.

The value '*Embrown*' can be deleted. But this value is only one essential value in t_6 and, therefore, t_6 can be deleted too. After that *splus*(*Blond*) is modified to the set {4,8}.

We observe that *t*({4,8}) = '*Tall Blond*' is a test. Hence, the value '*Blond*' can be deleted from further consideration together with the row t_4. Now the intersection of the rows t_5, t_7, and t_8 produces the test '*Tall Blue*'.

Table - 12. The Essential Values for the Examples *t_4, t_5, t_6, t_7, and t_8.*

Index of Example	Height	Color of Hair	Color of Eyes	Essential Values	Class
1	Short	Blond	Blue		1
2	Short	Brown	Blue		1
3	Tall	Brown	Embrown		1
4	Tall	Blond	Embrown	Blond	2
5	Tall	Brown	Blue	*Blue, Tall*	2
6	Short	Blond	Embrown	Embrown	2
7	Tall		Blue	Tall, Blue	2
8	Tall	Blond	Blue	Tall	2

An Approach for Searching for Essential Examples

Let *STGOOD* be the partially ordered set of elements *s* satisfying the condition that *t*(*s*) is a GMRT for *R*(+). We can use the set *STGOOD* to find indices of essential examples in some subset *s** of indices for which *t*(*s**) is not a test. Let *s** = {i_1, i_2, … , i_q}. Construct the set of intersections {*s** ∩ *s*': *s*'∈ *STGOOD*}. Any obtained intersection *corresponds* to a test for positive examples. Take one of the intersections with the maximal number of indices. The subset of *s** complementing in *s** the maximal intersection is the minimal set of indices of essential examples in *s**. For instance, *s** = {2,3,4,7,8}, *s*' = {2,3,4,7}, *s*' ∈ *STGOOD*, hence 8 is the index of essential example t_8 in *s**.

In the beginning of inferring GMRTs, the set *STGOOD* is empty. Next we describe the procedure with the use of which a quasi-maximal subset of *s** that corresponds to a test is obtained.

We begin with the first index i_1 of *s**, then we take the next index i_2 of *s** and evaluate the function to_be_test (*t*({i_1, i_2})). If the value of the function is *true*, then we take the next index i_3 of *s** and evaluate the function to_be_test (*t*({i_1, i_2, i_3})). If the value of the function to_be_test (*t*({i_1, i_2})) is *false*, then the index i_2 of *s** is skipped and the function to_be_test (*t*({i_1, i_3})) is evaluated. We continue this process until we achieve the last index of *s**.

For example, in Table 6, *S*(+) = {4,5,6,7,8}. Find the quasi-minimal subset of indices of essential examples for *S*(+). Using the procedure described above we get that *t*({4,6}) = '*Blond Embrown*' is a test for the second class and 5,7,8 are the indices of essential examples in S(+). Consider row t_5. We know that '*Blue*' is essential in it (see, please, Table 12). We have *t*(*splus*({*Blue*})) = *t*({5,7,8}) = '*Tall Blue*', and '*Tall Blue*' is a test for the second class of examples. Delete '*Blue*' and t_5. Now t_7 is not a test and we delete it. After that *splus*({*Tall*}) is modified to be the set {4,8}, and *t*({4,8}) = '*Tall Blond*' is a test. Hence, the value '*Tall*' together with row t_8 cannot be considered for searching for new tests. Finally *S*(+) = {4,6} corresponds to the test already known.

An Approach for Incremental Algorithms

The decomposition of the main problem of inferring GMRTs into subtasks of the first or second kind gives the possibility to construct incremental algorithms for this problem. The simplest way to do it consists of the following steps: choose example (value), form subproblem, solve subproblem (with the use of Algorithm 1 or Algorithm 2), delete example (value) after the subproblem is over, reduce *R*(+) and *T* and check the condition of ending the main task.

A recursive procedure for using attributive subproblems for inferring GMRTs has been described in [Naidenova et al., 1995b]. Some complexity evaluations of this algorithm can be found in [Naidenova and Ermakov, 2001]. In the following part of this chapter, we give an algorithm for inferring GMRTs the core of which is the decomposition of the main problem into the subtasks of the first kind combined with searching essential examples.

DIAGaRa: An Algorithm for Inferring All GMRTs with the Decomposition into Subtasks of the First Kind

The algorithm DIAGaRa for inferring all the GMRTs with the decomposition into subproblems of the first kind is briefly described in Figure 1.

The Basic Recursive Algorithm for Solving a Subtask of the First Kind

The initial information for the algorithm of finding all the GMRTs contained in a positive example is the projection of this example on the current set *R*(+). Essentially the projection is simply a subset of examples defined on a certain restricted subset *t** of values. Let *s** be the subset of indices of examples from *R*(+) which have produced the projection.

```
s* ← s(+) ={1,   , nt};
t* ← T;
Do
Begin
1.  to find all the GMRTs for a given set of positive examples with
    the use of the basic algorithm of solving subtask of the first
    kind;
End
```

Figure - 1. The Algorithm DIAGaRa.

It is useful to introduce the characteristic *W*(*t*) of any collection *t* of values named by the weight of *t* in the projection: *W*(*t*) = ||*s** ∩ *s*(*t*)|| is the number of positive examples of the projection containing *t*. Let *WMIN* be the minimal permissible value of the weight.

Let *STGOOD* be the partially ordered set of elements *s* satisfying the condition that *t*(*s*) is a good test for *R*(+).

The basic algorithm consists of applying the sequence of the following steps:

Step 1. Check whether the intersection of all the elements of projection is a test and if so, then *s** is stored in *STGOOD* if *s** corresponds to a good test at the current step; in this case the subtask is over. Otherwise the next step is performed (we use the function to_be_test(*t*): if *s*(*t*) ∩ *S*(+) = *s*(*t*) (*s*(*t*) ⊆ *S*(+)) then *true* else *false*).

Step 2. For each value *A* in the projection, the set *splus*(*A*) = {*s** ∩ *s*(*A*)} and the weight *W*(*A*) = ||*splus*(*A*)|| are determined and if the weight is less than the minimum permissible weight *WMIN*, then the value *A* is deleted from the projection. We can also delete the value *A* if *W*(*A*) is equal to *WMIN* and *t*(*splus*(*A*)) is not a test – in this case *A* will not appear in a maximally redundant test *t* with *W*(*t*) equal to or greater than *WMIN*.

Step 3. The generalization operation is performed: *t*′= *t*(*splus*(*A*)), *A* ∈ *t**; if *t*′is a test, then the value *A* is deleted from the projection and *splus*(*A*) is stored in *STGOOD* if *splus*(*A*) corresponds to a good test at the current step.

Step 4. The value *A* can be deleted from the projection if *splus*(*A*) ⊆ *s’* for some *s’* ∈ *STGOOD*.

Step 5. If at least one value has been deleted from the projection, then the reduction of the projection is necessary. The reduction consists of deleting the elements of projection that are not tests (as a result of previous eliminating values). If, under reduction, at least one element has been deleted from the projection, then Step 2, Step 3, Step 4, and Step 5 are repeated.

Step 6. Check whether the subtask is over or not. The subtask is over when either the projection is empty or the intersection of all elements of the projection corresponds to a test (see Step 1). If the subtask is not over, then the choice of an essential example in this projection is performed and the new subtask is formed with the use of this essential example. The new subsets s^* and t^* are constructed and the basic algorithm runs recursively. The important part of the basic algorithm is how to form the set *STGOOD.*

We give in the Appendix an example of the work of the algorithm DIAGaRa.

An Approach for Forming the Set *STGOOD*

Let $L(S)$ be the set of all subsets of the set S. $L(S)$ is the set lattice [Rasiova, 1974]. The ordering determined in the set lattice coincides with the set-theoretical inclusion. It will be said that subset s_1 is absorbed by subset s_2, i.e. $s_1 \leq s_2$, if and only if the inclusion relation is hold between them, i.e. $s_1 \subseteq s_2$. Under formation of *STGOOD*, a collection s of indices is stored in *STGOOD* if and only if it is not absorbed by any collection of this set. It is necessary also to delete from *STGOOD* all the collections of indices that are absorbed by s if s is stored in *STGOOD*. Thus, when the algorithm is over, the set *STGOOD* contains all the collections of indices that correspond to GMRTs and only such collections. Essentially the process of forming *STGOOD* is an incremental procedure of finding all maximal elements of a partially ordered set. The set *TGOOD* of all the GMRTs is obtained as follows: $TGOOD = \{t: t = t(s), (\forall s) (s \in STGOOD)\}$.

The Estimation of the Number of Subtasks to Be Solved

The number of subtasks at each level of recursion is determined by the number of essential examples in the projection associated with this level. The depth of recursion for any subtask is determined by the greatest cardinality (call it '*CAR*') of set-theoretical intersections of elements $s \in STGOOD$ corresponding to GMRTs: $CAR = \max (||s_i \cap s_j||, \forall(s_i, s_j)\ s_i, s_j \in STGOOD)$. In the worst case, the number of subtasks to be solved is of order $O(2^{CAR})$.

CASCADE: Inferring all GMRTs of Maximal Weight

The algorithm CASCADE serves for inferring all the GMRTs of maximal weight. At the beginning of the algorithm, the values are arranged in decreasing order of weight such that $W(A_1) \geq W(A_2) \geq \ldots \geq W(A_m)$, where $A_1, A_2, \ldots, A_m$ is a permutation of values. The shortest sequence of values $A_1, A_2, \ldots, A_j, j \leq m$ is defined such that it is a test for positive examples and *WMIN* is made equal to $W(A_j)$. The procedure DIAGaRa tries to infer all the GMRTs with weight equal to *WMIN*. If such tests are obtained, then the algorithm stops. If such tests are not found, then *WMIN* is decreased, and the procedure DIAGaRa runs again.

Conclusion

In this paper, we used a unified model for inferring implicative logical rules from examples. The key concept of our approach is the concept of a good diagnostic test. We define a good diagnostic test as the best approximation of a given classification on a given set of examples. In the framework of our approach, we show the equivalence between implicative rules and diagnostic tests for a given set of examples. The task of inferring good diagnostic tests from examples serves as an ideal model of inductive reasoning because this task realizes the canons of induction that has been originally formulated by English logician J.-S. Mill.

We have given the decomposition of inferring all good maximally redundant tests for a given set of examples into operations and subtasks that are in accordance with main human common sense reasoning operations. This decomposition allows, in principle, to transform the process of inferring good tests (and implicative rules) into a “step by step” reasoning process. Incremental algorithms of inferring good classification tests from examples demonstrate the possibility of this transformation in the best way.

We consider two kinds of subtasks: for a given set of positive examples 1) given a positive example t, find all GMRTs contained in t; 2) given a non-empty collection of values X (maybe only one value) such that it is not a test, find all GMRTs containing X. The decomposition of good classification tests inferring into subtasks implies introducing a set of special rules to realize the following operations: choosing examples (values) for subtasks, forming subtasks, deleting values or examples from subtasks and some other rules controlling the process of good test inferring. The concepts of an essential value and an essential example are introduced in order to optimize the choice of subtasks of the first and second kinds.

We have described an inductive algorithm DIAGaRa for inferring all good maximally redundant tests for a given set of positive examples. This algorithm realizes one of the possibilities to transform the searching of diagnostic tests (implicative logical rules) into “step by step” learning procedure.

Our approach is also applicable for inferring functional and associative dependencies from data.

Acknowledgements

The author is very grateful to Professor Evangelos Triantaphyllou (Louisiana StateUniversity) who inspired and supported this paper, and to Dr. Giovanni Felici (IASI – Italian National Research Council), for his invaluable advice concerning all the parts of this work.

Appendix

The data to be processed are in Table 13 (the set of positive examples) and in Table 14 (the set of negative examples).

An Example of Using the Algorithm DIAGaRa

We use the algorithm DIAGaRa for inferring all the GMRTs having the weight equal to or greater than *WMIN* = 4 for the training set of examples represented in Table 13 (the set of positive examples) and in Table 14 (the set of negative examples).

We begin with $s^* = S(+) = \{\{1\}, \{2\}, \ldots, \{14\}\}$, $t^* = T = \{A_1, A_2, \ldots.., A_{26}\}$, $SPLUS = \{splus(A_i): A_i \in t^*\}$ (see *SPLUS* in Table 15).

In table 15 and 16, A_* denotes the collection of values $\{A_8\ A_9\}$ and A_+ denotes the collection of values $\{A_{14}\ A_{15}\}$ because $splus(A_8) = splus(A_9)$ and $splus(A_{14}) = splus(A_{15})$.

Please observe that $splus(A_{12}) = \{2,3,4,7\}$ and $t(\{2,3,4,7\})$ is a test, therefore, A_{12} is deleted from t^* and $splus(A_{12})$ is inserted into *STGOOD*. Then $W(A_*)$, $W(A_{13})$, and $W(A_{16})$ are less than *WMIN*, hence we can delete A_*, A_{13}, and A_{16} from t^*. Now t_{10} is not a test and can be deleted. After modifying $splus(A)$ for A_5, A_{18}, A_2, A_3, A_4, A_6 A_{20}, A_{21}, and A_{26} we find that $W(A_5) < WMIN$, therefore, A_5 is deleted from t^* and $splus(A_5)$ is inserted into *STGOOD*. Then $W(A_{18})$ turns out to be less than *WMIN* and we delete A_{18}, which implies deleting t_{13}. Next we modify $splus(A)$ for A_1, A_{19}, A_{23}, A_4, A_{26} and find that $splus(A_4) = \{2,3,4,7\}$. A_4 is deleted from t^*. Finally, $W(A_1)$ turns out to be less than *WMIN* and we delete A_1.

Table - 13. The Set of Positive Examples *R*(+).

Index of example	*R*(+)
1	$A_1\ A_2\ A_5\ A_6\ A_{21}\ A_{23}\ A_{24}\ A_{26}$
2	$A_4\ A_7\ A_8\ A_9\ A_{12}\ A_{14}\ A_{15}\ A_{22}\ A_{23}\ A_{24}\ A_{26}$
3	$A_3\ A_4\ A_7\ A_{12}\ A_{13}\ A_{14}\ A_{15}\ A_{18}\ A_{19}\ A_{24}\ A_{26}$
4	$A_1\ A_4\ A_5\ A_6\ A_7\ A_{12}\ A_{14}\ A_{15}\ A_{16}\ A_{20}\ A_{21}\ A_{24}\ A_{26}$
5	$A_2\ A_6\ A_{23}\ A_{24}$
6	$A_7\ A_{20}\ A_{21}\ A_{26}$
7	$A_3\ A_4\ A_5\ A_6\ A_{12}\ A_{14}\ A_{15}\ A_{20}\ A_{22}\ A_{24}\ A_{26}$
8	$A_3\ A_6\ A_7\ A_8\ A_9\ A_{13}\ A_{14}\ A_{15}\ A_{19}\ A_{20}\ A_{21}\ A_{22}$
9	$A_{16}\ A_{18}\ A_{19}\ A_{20}\ A_{21}\ A_{22}\ A_{26}$
10	$A_2\ A_3\ A_4\ A_5\ A_6\ A_8\ A_9\ A_{13}\ A_{18}\ A_{20}\ A_{21}\ A_{26}$
11	$A_1\ A_2\ A_3\ A_7\ A_{19}\ A_{20}\ A_{21}\ A_{22}\ A_{26}$
12	$A_2\ A_3\ A_{16}\ A_{20}\ A_{21}\ A_{23}\ A_{24}\ A_{26}$
13	$A_1\ A_4\ A_{18}\ A_{19}\ A_{23}\ A_{26}$
14	$A_{23}\ A_{24}\ A_{26}$

Table - 14. The Set of Negative Examples *R*(–).

Index of example	*R*(–)	Index of example	*R*(–)
15	A_3 A_8 A_{16} A_{23} A_{24}	32	A_1 A_2 A_3 A_7 A_9 A_{10} A_{11} A_{13} A_{18}
16	A_7 A_8 A_9 A_{16} A_{18}	33	A_1 A_5 A_6 A_8 A_9 A_{10} A_{19} A_{20} A_{22}
17	A_1 A_{21} A_{22} A_{24} A_{26}	34	A_2 A_8 A_9 A_{18} A_{20} A_{21} A_{22} A_{23} A_{26}
18	A_1 A_7 A_8 A_9 A_{13} A_{16}	35	A_1 A_2 A_4 A_5 A_6 A_7 A_9 A_{13} A_{16}
19	A_2 A_6 A_7 A_9 A_{21} A_{23}	36	A_1 A_2 A_6 A_7 A_8 A_{10} A_{11} A_{13} A_{16} A_{18}
20	A_{10} A_{19} A_{20} A_{21} A_{22} A_{24}	37	A_1 A_2 A_3 A_4 A_5 A_6 A_7 A_{12} A_{14} A_{15} A_{16}
21	A_1 A_{10} A_{20} A_{21} A_{22} A_{23} A_{24}	38	A_1 A_2 A_3 A_4 A_5 A_6 A_9 A_{11} A_{12} A_{13} A_{16}
22	A_1 A_3 A_6 A_7 A_9 A_{10} A_{16}	39	A_1 A_2 A_3 A_4 A_5 A_6 A_{14} A_{15} A_{19} A_{20} A_{23} A_{26}
23	A_2 A_6 A_8 A_9 A_{14} A_{15} A_{16}	40	A_2 A_3 A_4 A_5 A_6 A_7 A_{11} A_{12} A_{13} A_{14} A_{15} A_{16}
24	A_1 A_4 A_5 A_6 A_7 A_8 A_{11} A_{16}	*41*	A_2 A_4 A_5 A_6 A_7 A_9 A_{10} A_{11} A_{12} A_{13} A_{14} A_{15} A_{19}
25	A_7 A_{10} A_{11} A_{13} A_{19} A_{20} A_{22} A_{26}	42	A_1 A_2 A_3 A_4 A_5 A_6 A_{12} A_{16} A_{18} A_{19} A_{20} A_{21} A_{26}
26	A_1 A_2 A_3 A_5 A_6 A_7 A_{10} A_{16}	43	A_4 A_5 A_6 A_7 A_8 A_9 A_{10} A_{11} A_{12} A_{13} A_{14} A_{15} A_{16}
27	A_1 A_2 A_3 A_5 A_6 A_{10} A_{13} A_{16}	44	A_3 A_4 A_5 A_6 A_8 A_9 A_{10} A_{11} A_{12} A_{13} A_{14} A_{15} A_{18} A_{19}
28	A_1 A_3 A_7 A_{10} A_{11} A_{13} A_{19} A_{21}	45	A_1 A_2 A_3 A_4 A_5 A_6 A_7 A_8 A_9 A_{10} A_{11} A_{12} A_{13} A_{14} A_{15}
29	A_1 A_4 A_5 A_6 A_7 A_8 A_{13} A_{16}	46	A_1 A_3 A_4 A_5 A_6 A_7 A_{10} A_{11} A_{12} A_{13} A_{14} A_{15} A_{16} A_{23} A_{24}
30	A_1 A_2 A_3 A_6 A_{11} A_{12} A_{14} A_{15} A_{16}	47	A_1 A_2 A_3 A_4 A_5 A_6 A_8 A_9 A_{10} A_{11} A_{12} A_{14} A_{16} A_{18} A_{22}
31	A_1 A_2 A_5 A_6 A_{11} A_{14} A_{15} A_{16} A_{26}	48	A_2 A_8 A_9 A_{10} A_{11} A_{12} A_{14} A_{15} A_{16}

Table - 15. The Set *SPLUS* of the Collections *splus*(*A*) for all *A* in Tables 13 and 14.

SPLUS = {*splus*(A_i): *s*(A_i) ∩ *S*(+), A_i ∈ *T*}:	
splus(A_*) → {2,8,10}	*splus*(A_{22}) → {2,7,8,9,11}
splus(A_{13}) → {3,8,10}	*splus*(A_{23}) → {1,2,5,12,13,14}
splus(A_{16}) → {4,9,12}	*splus*(A_3) → {3,7,8,10,11,12}
splus(A_1) → {1,4,11,13}	*splus*(A_4) → {2,3,4,7,10,13}
splus(A_5) → {1,4,7,10}	*splus*(A_6) → {1,4,5,7,8,10}
splus(A_{12}) → {2,3,4,7}	*splus*(A_7) → {2,3,4,6,8,11}
splus(A_{18}) → {3,9,10,13}	*splus*(A_{24}) → {1,2,3,4,5,7,12,14}
splus(A_2) → {1,5,10,11,12}	*splus*(A_{20}) → {4,6,7,8,9,10,11,12}
splus(A_+) → {2,3,4,7,8}	*splus*(A_{21}) → {1,4,6,8,9,10,11,12}
splus(A_{19}) → {3,8,9,11,13}	*splus*(A_{26}) → {1,2,3,4,6,7,9,10,11,12,13,14}

Table - 16. The sets *STGOOD* and *TGOOD* for the Examples of Tables 13 and 14.

№	*STGOOD*	*TGOOD*
1	2,3,4,7	A_4 A_{12} A_+ A_{24} A_{26}
2	1,2,12,14	A_{23} A_{24} A_{26}
3	4,6,8,11	A_7 A_{20} A_{21}

We can delete also the values A_2, A_{19} because $W(A_2)$, $W(A_{19})$ = 4, $t(splus(A_2))$, $t(splus(A_{19}))$ are not tests and, therefore, these values will not appear in a maximally redundant test *t* with $W(t)$ equal to or greater than 4.

After deleting these values we can delete the examples t_9, t_5 because A_{19} is essential in t_9, and A_2 is essential in t_5. Next we can observe that $splus(A_{23})$ = {1,2,12,14} and *t*({1,2,12,14}) is a test, thus A_{23} is deleted from t^* and $splus(A_{23})$ is inserted into *STGOOD*. We can delete the value A_{22} and A_6 because $W(A_{22})$ and $W(A_6)$ are now equal to 4, $t(splus(A_{22}))$ and $t(splus(A_6))$ are not tests and these values will not appear in a maximally redundant test with weight equal to or greater than 4. Now t_{14} and t_1 are not tests and can be deleted.

Now choose t_6 as a subtask because this positive example is more difficult to be distinguished from the negative examples. By resolving this subtask, we find that t_6 produces a new test *t* with *s*(*t*) equal to {4,6,8,11}. Delete t_6. We can also delete the value A_{21} because $W(A_{21})$ is now equal to 4, $t(splus(A_{21}))$ is not a test and this value will not appear in a maximally redundant test with weight equal to or greater than 4.

Now choose t_8 as a subtask because it belongs to the set of essential examples in the current projection with respect to the subset {2,3,4,7} that corresponds to one of the GMRTs already obtained. By resolving this subtask, we find that t_8 does not produce any new test. Delete $t_{8.}$ After that we can delete the values A_+, A_7, A_3, and A_{20} and these deletions imply than all of the remaining rows t_2, t_3, t_4, t_7, t_{11}, and t_{12} are not tests.

The list of all the GMRTs for the training set of positive examples is given in Table 16.

Bibliography

[Boldyrev, 1974]N. G. Boldyrev, "Minimization of Boolean Partial Functions with a Large Number of "Don't Care" Conditions and the Problem of Feature Extraction", *Proceedings of International Symposium "Discrete Systems"*, Riga, Latvia, pp.101-109, 1974.

[Cosmadakis et al., 1986] S. Cosmadakis, P. C. Kanellakis, N. Spyratos, "Partition Semantics for Relations", *Journal of Computer and System Sciences*, Vol. 33, No. 2, pp.203-233, 1986.

[Demetrovics and Vu, 1993] J. Demetrovics and D. T. Vu, "Generating Armstrong Relation Schemes and Inferring Functional Dependencies from Relations", *International Journal on Information Theory & Applications*, Vol. 1, No. 4, pp.3-12, 1993.

[Finn, 1984] V. K. Finn, "Inductive Models of Knowledge Representation in Man-Machine and Robotics Systems", *Proceedings of VINITI*, Vol. A, pp.58-76, 1984.

[Ganascia, 1989] J.- Gabriel. Ganascia, "EKAW - 89 Tutorial Notes: Machine Learning", *Third European Workshop on Knowledge Acquisition for Knowledge-Based Systems*, Paris, France, pp. 287-296, 1989.

[Huntala et al., 1999] Y. Huntala, J. Karkkainen, P. Porkka, and H. Toivonen, "TANE: An Efficient Algorithm for Discovering Functional and Approximate Dependencies", *The Computer Journal*, Vol. 42, No. 2, pp. 100-111, 1999.

[Kuznetsov, 1993] S. O. Kuznetsov, "Fast Algorithm of Constructing All the Intersections of Finite Semi-Lattice Objects", *Proceedings of VINITI*, Series 2, No. 1, pp. 17-20, 1993.

[Mannila and Räihä, 1992] H. Mannila, and K. – J. Räihä, "On the Complexity of Inferring Functional Dependencies", *Discrete Applied Mathematics*, Vol. 40, pp. 237-243, 1992.

[Mannila and Räihä, 1994] H. Mannila, and K. – J. Räihä, "Algorithm for Inferring Functional Dependencies". *Data* & *Knowledge Engineering*, Vol. 12, pp. 83-99, 1994.

[Megretskaya, 1989] I. A. Megretskaya, "Construction of Natural Classification Tests for Knowledge Base Generation", in: *The Problem of the Expert System Application in the National Economy*, Kishinev, Moldavia, pp. 89-93, 1988.

[Mill, 1900] J. S. Mill, *The System of Logic*, Russian Publishing Company "Book Affair": Moscow, Russia, 1900.

[Naidenova and Polegaeva, 1986] X. A. Naidenova, J. G. Polegaeva, "An Algorithm of Finding the Best Diagnostic Tests", *The 4-th All Union Conference "Application of Mathematical Logic Methods"*, Theses of Papers, Mintz, G; E, Lorents, P. P. (Eds), Institute of Cybernetics, National Acad. of Sciences of Estonia, Tallinn, Estonia, pp. 63-67, 1986.

[Naidenova and Polegaeva, 1991] X. A. Naidenova, J. G. Polegaeva, "The System of Knowledge Acquisition from Experimental Facts", in: *"Industrial Applications of Artificial Intelligence*", James L. Alty and Leonid I. Mikulich (Eds), Elsevier Science Publishers B.V., Amsterdam, The Netherlands, pp. 87-92, 1991.

[Naidenova, 1992] X. A. Naidenova, "Machine Learning As a Diagnostic Task", in: *"Knowledge-Dialogue-Solution", Materials of the Short-Term Scientific Seminar*, Saint-Petersburg, Russia, editor Arefiev, I., pp.26-36, 1992.

[Naidenova et al., 1995a] X. A. Naidenova, J. G. Polegaeva, J. E. Iserlis, "The System of Knowledge Acquisition Based on Constructing the Best Diagnostic Classification Tests", *Proceedings of International Conference "Knowledge-Dialog-Solution"*, Jalta, Ukraine, Vol. 1, pp. 85-95, 1995a.

[Naidenova et al., 1995b] X. A. Naidenova, M. V. Plaksin, V. L. Shagalov, "Inductive Inferring All Good Classification Tests", *Proceedings of International Conference "Knowledge-Dialog-Solution*", Jalta, Ukraine, Vol. 1, pp.79-84, 1995b.

[Naidenova, 1996] X. A. Naidenova, "Reducing Machine Learning Tasks to the Approximation of a Given Classification on a Given Set of Examples", *Proceedings of the 5-th National Conference at Artificial Intelligence*, Kazan, Tatarstan, Vol. 1, pp. 275-279, 1996.

[Naidenova, 1999] X. A. Naidenova, "The Data-Knowledge Transformation", in: "*Text Procesing and Cognitive Technologies", Paper Collection*, editor Solovyev, V. D., - Pushchino, Russia, Vol. 3, pp. 130-151, 1999.

[Naidenova and Ermakov, 2001] X. A. Naidenova, A. E. Ermakov, "The Decomposition of Algorithms of Inferring Good Diagnostic Tests", *Proceedings of the 4-th International Conference "Computer – Aided Design of Discrete Devices" (CAD DD'2001)*, Institute of Engineering Cybernetics, National Academy of Sciences of Belarus, editor A. Zakrevskij, Minsk, Belarus, Vol. 3, pp. 61-69, 2001.

[Naidenova, 2001] X. A. Naidenova, "Inferring Good Diagnostic Tests as a Model of Common Sense Reasoning", *Proceedings of the International Conference "Knowledge-Dialog-Solution" (KDS'2001)*, State North-West Technical University, Publishing House « Lan », Saint-Petersburg, Russia, Vol. II, pp. 501-506, 2001.

[Ore, 1944] O. Ore, "*Galois Connexions*", Trans. Amer. Math. Society, Vol. 55, No. 1, pp. 493-513, 1944.

[Piaget, 1959] J.Piaget, *La genèse des Structures Logiques Elémentaires*, Neuchâtel, 1959.

[Riguet, 1948] J. Riguet, “Relations Binaires, Fermetures, Correspondences de Galois”, *Bull. Soc. Math*., France, Vol. 76., No 3, pp.114-155, 1948.

[Shreider, 1974] J. Shreider, “Algebra of Classification”, *Proceedings of VINITI*, Series 2, No. 9, pp. 3-6, 1974.

[Sperner, 1928] E. Sperner, “Eine satz uber Untermengen einer Endlichen Menge”. *Mat. Z*., Vol. 27, No. 11, pp. 544-548, 1928.

[Wille, 1992] R. Wille, “Concept Lattices and Conceptual Knowledge System”, *Computer Math. Appl*., Vol. 23, No. 6-9, pp. 493-515, 1992.

Author's Information

Naidenova Xenia Alexandrovna - Military medical academy, Saint-Petersburg, Stoikosty street, 26-1-248, naidenova@mail.spbnit.ru.

ACTIVE MONITORING AND DECISION MAKING PROBLEM

Sergey Mostovoi, Vasiliy Mostovoi

Abstract: Active monitoring and problem of non-stable of sound signal parameters in the regime of piling up response signal of environment is under consideration. Math model of testing object by set of weak stationary dynamic actions is offered. The response of structures to the set of signals is under processing for getting important information about object condition in high frequency band. Making decision procedure by using researcher’s heuristic and aprioristic knowledge is discussed as well. As an example the result of numerical solution is given.

Keywords: math model, active monitoring, set of weak stationary dynamic actions.

ACM Classification Keywords: I.6.1 Simulation Theory.

Introduction

The distinctive feature of seismic monitoring is the particular, seismic frequency range, encompassing infrasonic and low range of a sound spectrum. The characteristics of each monitoring object are slowly varied in time, but at the same time sometimes processes might be occurred is too rapid. The seismic monitoring deals with the large size objects, down to the sizes of a terrestrial Globe. Because of mankind anxiety on possible earthquakes, the extremely passive monitoring has a deep history, but at latest time the active monitoring is often used. The active monitoring is such an experiment, which one is connected to generation of sounding signal of a different type, both on a spectral band, and on duration and power, down to atomic explosions. But in active experiment only monitoring approach enables to obtain ecological pure result, i.e. without any of appreciable influencing on an environment. Monitoring is a set of regime observations, and condition of observations and the characteristics of sounding signal depend on the purposes of given investigation. There are many such purposes, but, from our point of view, we select two basic one. It is dynamics of variations happening in investigated object, and it is detail of estimations, which characterise this object. Despite of large discrepancy of these two purpose, the approaches both to experimentation and to processing receivable data are very close, as well as problems, originating at it.

To problems, first of all from the ecological point of view, it is necessary to refer necessity to realize active monitoring of investigated object by low-power signals, commensurable with a level of a natural background. This circumstance results that the estimation of sounding signal parameters, passing the studied object, i.e. signal response of an investigated system on a sounding signal, is hampered because of a low signal-noise proportion.

Therefore there is a necessity for the special conditions of experiment and applying special, sometimes very composite, signal processing procedures of an investigated system response. The used above words "the regime observations" consider rigid stability in implementation of a condition. It means stability of monitoring time characteristic and parameter stability of a sounding signal, i.e. invariance of its spectral characteristic. With evidence it is clear, that always there is an extreme accuracy of arguments describing a signal and arguments temporary experiment providing. In this article the problem is put: when and to what arguments the instability is essential, in what it results, and how to eliminate its influencing, if it is possible?

First of all, it is necessary to construct a mathematical model of experiment, in which one the most essential moments of monitoring process would be reflected, including both processes, and, accompanying this process background noise, and natural hum noise. The prior knowledge of noise stochastic process will allow largely weakening its influencing on deriving of estimation obtainment of process arguments, which one is perceived as a useful signal. This slackening is reached by optimization of processing procedures, which is taking into account prior statisticians of noise stochastic processes.

In a series of treatises [1-4] the separate aspects of a reduced problem were regarded. Into the given paper there is an attempt to summarize earlier reviewed the approach to procedure modeling of active experiment, analysis of experiment parameters instability influence and optimization of procedure processing of observed data, by yardsticks taking into account the characteristics of a natural background noise, instability of sounding signal parameters and consequences caused by this instability.

The Mathematical Model of Active Monitoring

The math model of i-th experiment in a serial from M -th ones is proposed. In active monitoring serial can be introduced as follows:

$$y_i(t) = S\left(t, \tau_i, \vec{h}_i\right) * H(t) + n_i(t), \quad t \in \left(\tau_i, \tau_i + T\right), \tag{1}$$

Where i is number of experiment, $y_i(t)$ - response of environment to an sound signal $S\left(t, \tau_i, \vec{h}_i\right)$, depending from vector of parameters $\vec{h}_i$, which one is convoluted with reacting of environment $H(t)$ on a delta-function signal $\delta(t)$, $n_i(t)$ an additive noise accompanying experiment, T - duration of one experiment, $\left(\tau_i, \tau_i + T\right)$ - time period of i -th experiment conducting of, and $*$ - a convolution operator symbol. The experiment is constructed in such a manner that energy of a signal, registered by sensors, $E\left[\ S\left(t, \vec{h}_i\right) * H(t)\ \right]$ and energy of a natural background $E\left[n_i(t)\right]$ are commensurable in the selected metric, it means, that influencing of experiment on a state of the environment is negligible. In the pattern that circumstance is taken into account, that the non-linear phenomena in experiment can be neglected, a linear routine of the specification statement of interplay of environment and exploring signal by the way convolutions therefore is selected. Let's mark, that the convolution is described by following integral:

$$S\ (t) * H(t) = \int_0^{\infty} H(\tau) S\ (t - \tau) d\tau, \tag{2}$$

The full experiment is defined by following model

$$y(t) = \sum_{i=1}^{M} y_i(t) \tag{3}$$

As a time of experiment T we shall consider the time for which one the reaction level of environment to an exploring signal becomes less then some level ε, which one can be selected depending on a level of a natural background. For example, in the metric $C_{(\tau_i + T, \tau_i + \Gamma)}$; $\Gamma >> T$ is instituted from a condition:

$$\max(y_i(t)) \le \varepsilon \text{ ; } t \in (\tau_i + T, \tau_i + \Gamma) \text{ for } \forall\ \tau_i \tag{4}$$

Certainly ε, and after it and T as well, is exclusively selected by the feeling of explorer heuristics, his point of view to experiment and a priori estimations of a noise $n(t)$ power. As it was noted, the monitoring guesses a serial from M experiments, i.e. $i = \overline{1, M}$.

The Model of an Exploring Signal

Let's consider, that the signal $S(t, \vec{h}_i)$ depends on the vector of parameters $\vec{h}_i = \{h_{i1}, ..., h_{iN}\}$, which components are define the shape and energy of signal. It is naturally to consider that a signal is physically realizable, i.e. to be fitting two conditions: causality and stability. The same conditions are natural to the reacting of the environment $H(t)$ as well.

$$S\ (t, \vec{h}_i) = \begin{Bmatrix} S\ (t, \vec{h}_i), t \ge 0 \\ 0, t < 0 \end{Bmatrix}; \qquad \int\limits_0^{\infty} (S\ (t, \vec{h}_i))^2\, dt < \infty \tag{5}$$

Causality means, that if the signal has been started at the moment τ_i, it means that the experiment has begun at this moment and up to this moment the signal did not exist.

$$S\ (t, \vec{h}_i, \tau_i) = \begin{Bmatrix} S\ (t - \tau_i, \vec{h}_i), t - \tau_i \ge 0 \\ 0, t - \tau_i < 0 \end{Bmatrix} \tag{6}$$

In a condition of causality we at once consider also a condition of stationarity that is reflected in the dependence of a signal on a difference of time t and the signal start moment τ_i

The stability means, that for any value ε of an energy level in the metric L_2 there is such value of T, that

$$\int\limits_T^{\infty} (S\ (t, \vec{h}_i))^2\, dt < \varepsilon \text{. for } \forall\ h_i \tag{7}$$

The last circumstance allows to determine duration of one experiment T, for this purpose it is necessary, that the level ε was less or much less then the energy level of a natural background.

It is possible to consider τ_i as one of the component (for example, with a zero subscript) of a vector of arguments, which are defining the signal and which are non-linear - including in the pattern of a signal. The duration value of T is a value of deterministic argument, for example, which is equal to the last component of vector $\vec{h}$. Let's try to represent other non-linear arguments of a signal. The signal can be introduced as a linear combination of known functions (for example, fragment of a vector of orthogonal functions $\{\varphi_k(t - \tau_i, k \cdot \omega_{0i})\chi(t, \tau_i - \psi_i, \tau_i + T)\}$, $k = \overline{1, N}$ at an interval of length T.

$$S\ (t, \vec{h}_i, \tau_i) = \sum_{k=1}^{N} h_{ik}\varphi_k(t - \tau_i, k \cdot \omega_{0i})\chi(t, \tau_i - \psi_i, \tau_i + T) \tag{8}$$

Here is ω_{0i} - a sample unit of random argument ω_0, which defines system of functions $\vec{\varphi}(t, \tau, \omega, \psi)$, and ψ is the applicable phase for this system

$$\vec{\varphi}(t, \tau, \omega, \psi) = \{\varphi_k(t - \tau, \omega \cdot k) \cdot \chi(t, \tau - \psi, \tau + T)\}, \quad k = 1, ..., N \quad . \tag{9}$$

Here is characteristic interval function $\chi(t, \tau - \psi, \tau + T)$, which is also a non-linear characteristic of the signal model, as well as argument T,

$$\chi(t,\tau_i-\psi_i,\tau_i+T)=\begin{Bmatrix}1, & t\in(\tau_i-\psi_i,\tau_i+T),\\ 0, & t\notin(\tau_i-\psi_i,\tau_i+T);\end{Bmatrix}.$$

Let's consider argument ω_0 as one more component of arguments vector $\vec{h}$, namely h_{N+1}. Then ψ - h_{N+2}, and T we shall consider as a h_{N+3} component of vector $\vec{h}=\{h_k\}, k=0,...,N+3$.

In this case a sound signal in experiment with number i will be $S\ (t,\vec{h}_i)$.

So, the signal model is a random function which is supposed to be physically realizable and a stationary, which one is completely instituted by a random vector $\vec{h}$, N parameters of which one are linearly entered into the model.

Under consideration is a case, when set of vectors $\vec{h}_1,...,\vec{h}_M,$ are sampling from set of probable values of vector $\vec{h}$ with a priori known distribution $P(\vec{h})$. It means, that the stochastic nature of process $y(t)=\sum_{i=1}^{M}y_i(t)$ is defined by a random vector $\vec{h}$ and stochastic additive noise $n(t)$. As a determined component into this process is a response of environment $H(t)$ on a testing signal such as a delta-function. This response contains the environment information. As fluctuations of arguments of an exploring signal is determined and linearly, through the convolution equations, are connected to a signal $s(t,\vec{h}_i\ )$, registered by sensors on an exit of an observation system, that, allowing identifications (2) for a convolution $*$, we shall obtain

$s(t,\vec{h}_i\ )=S(t,\vec{h}_i)*H(t)$ and

$$y_i(t)=s(t,\vec{h}_i)+n_i(t),\quad t\in(\tau_i,\tau_i+T),\tau_i=h_{i0},T=h_{N+3}. \tag{10}$$

Hereinafter we shall esteem only response of environment $s(t,\vec{h}_i)$. Let's decipher separated values of a vector of components, defining both signal $S(t,\vec{h}_i)$ and response of environment $s(t,\vec{h}_i)$. First of all, try to separate arguments which are included linearly and non-linear into the model.

$$S(t,\vec{h}_i)=\left(\sum_{k=1}^{N}h_{ik}\varphi_k(t-h_{i0},h_{i,N+1}\cdot k)\right)\cdot\chi(t,h_{i0}-h_{i,N+2},h_{i0}+h_{iN+3}) \tag{11}$$

τ_{i0} is the component of vector $\vec{h}_i$ with zero index, ω_{i0} - $N+1$, and T - $N+2$ -th of a component. Function vector $\vec{\varphi}(t,\vec{h})=\{\varphi_k(t-\tau,\omega\cdot k)\cdot\chi(t,\tau-\psi,\tau+T)\},\quad k=1,...,N$ (12)

might be set of convenient for approximation an exploring cue of functions or piece orthonormalized on a spacing $(0,T)$ of basis functions. The approximating of an exploring signal in seismic survey by the way of damped sine wave can be regarded as the example

$$S(t,\vec{h}_i)=\theta_i\cdot\exp\{-\alpha_i t\}\cdot\sin\{\omega_i\cdot(t-\tau_i)\}\cdot\chi(t,\tau_i-\psi_i,\tau_i+T) \tag{13}.$$

In this case vector of free parameters of the pattern, which defines the signal, is $\vec{h}_i=\{h_{ik}\}=\{\tau_i,\theta_i,\alpha_i,\omega_i,\psi_i,T\},\quad k=0,...,5$ and has only five components, from which only the second one h_{i1} is entered into the model linearly. In general, and relevant for practice of seismic sounding case, the signal is represented by the way of approximating piece of its expansion in a series of orthonormalized base, as in the expression (6). The response of environment in i th experiment will be

$$s(t,\vec{h}_i)=\left(\sum_{k=1}^{N}h_{ik}\left(\int_{\tau_i}^{\tau_i+T}\varphi_k(t-\tau_i-\tau,\omega_i\cdot k)\cdot H(\tau)d\tau\right)\right)\cdot\chi(t,\tau_i-\psi_i,\tau_i+T) \tag{14}$$

Taking into account above mentioned result of a serial from M trials $y(t)$ becomes:

$$y(t)=\sum_{i=1}^{M} y_i(t)=\sum_{i=1}^{M} S\left(t,\vec{h}_i\right)*H(t)+n_i(t)=\sum_{i=1}^{M}\left(\sum_{k=1}^{N} h_{ik}\left(\int_{\tau_i}^{\tau_i+T}\varphi_k(t-\tau_i-\tau,\omega_i\cdot k)\cdot H(\tau)d\tau\right)\right)$$
$$\cdot\chi\left(t,\tau_i-\psi_i,\tau_i+T\right)+n_i(t),\quad t\in\left(0,M\cdot T\right) \tag{15}$$

Let's define:

$$\widetilde{\varphi}_k(t-\tau_i,\omega_i\cdot k,\psi_i)=\chi\left(t,\tau_i-\psi_i,\tau_i+T\right)\int_{\tau_i}^{\tau_i+T}\varphi_k(t-\tau_i-\tau,\omega_i\cdot k)\cdot H(\tau)d\tau\,. \tag{16}$$

With allowance for (16) model of monitoring becomes:

$$y(t)=\sum_{i=1}^{M}\left(\sum_{k=1}^{N} h_{ik}\widetilde{\varphi}_k(t-\tau_i,\omega_i\cdot k,\psi_i)\right)+n_i(t),\quad t\in\left(0,M\cdot T\right) \tag{17}$$

Model of Additive Noise $n(t)$

In this place we shall note, that for the further analysis the aprioristic knowledge of statistical characteristics of noise $\widetilde{n}(t)$ is important. The ideal situation is a knowledge of aprioristic distributions of all sections of stochastic process $\widetilde{n}(t)$, but in our case would be enough to know only its first moment $E[n(t)]=\mu(t)$, as the further procedure of processing assumes summation of record result of fragments of an experiment, i.e. reception of an estimation $\hat{E}[n(t)]$. This knowledge is still important and that as a result of carrying out of experiment and at data processing supervision there would be no accumulation of a regular error. The aprioristic knowledge of $\mu(t)$ will allow to carry out preliminary such procedure as $y(t)-\mu(t)$ and by that to minimize a regular error at an estimation of a signal, i.e. to take under consideration such process $n(t)$ for which $\mu(t)=0$. In this case procedure of summation of experiments set would state an estimation of value μ in each point *t* asymptotically, by quantity of the experiments, coming nearer to zero, i.e.

$$E[n(t)]=\mu(t) \tag{18}$$

Model of Data Processing

The following procedure of the observant data processing, which is based on piling signals up experiment of the environment response, is chosen.

$$\hat{E}[s(t)+n(t)]=\frac{1}{I}\sum_{i=1}^{M} y_i(t-(i-1)\cdot T)=\frac{1}{I}\sum_{i=1}^{I} s\left(t-(i-1)\cdot T,\vec{h}_i\right)+\frac{1}{I}\sum_{i=1}^{I} n\left(t-(i-1)\cdot T\right) \tag{19}$$

Here $\hat{E}[s(t)+n(t)]$ is an estimation of a population mean of the environment response and an additive noise, and $I=M\cdot T$ is the time of monitoring.

Density of distribution of the random parameters which are included in model (1) are necessary for definition of a population mean (19) us. Reception of estimations of aprioristic distributions of a vector of parameters $\vec{h}$ does not represent work since the source of probing signals always can be tested a priori, before carrying out of experiment, and the necessary statistics of the non-stable parameters determining a signal, thus can be received a priori.

We shall consider, that the aprioristic statistics gives the good consent with some density $dP\left(\vec{h}\right)$.

Example

With the purpose to get the monument spectral characteristics, logarithmic decrement of the oscillations of the object and to analyses of damping ability of the system, which was realized at the monument for oscillation reduction, the site tests were carried out. For registration of fluctuations three-directional geophone with gauges located on three mutually perpendicular axes was used. The special characteristics of gauges represent one-modal curve with the extreme point in f=l Hz. Geophones were placed at a horizontal surface, on the level of 42 meters. They served as a part of interface of the monitoring registration and processing automated system. This system allows correcting the spectral characteristic up to uniform in the chosen range of frequencies. The first part of experiment consisted in registration of monument reaction on a natural background as an input signal. This signal represents a superposition of the large number of the external factors from natural microseism noise and men made one up to signals from ground transport. The important moment is that the total spectrum of this signals is much wider then the response spectrum of the monument. For the monument it was obtained three modes on frequencies 0.48 Hz, 0.93 Hz and 1.47 Hz with corresponding amplitudes 1.0, 0.07 and 0.12. The frequency of 1.47 Hz with rather intensive amplitudes hypothetically is devoted to the mode of the top sculpture, the framework of which is less rigid then the framework of the self column. The second part of experiment was consisted in to get a logarithmic decrement of oscillation of the monument on the basic resonant frequency. For this purpose was used a damp of pendulum type. By compulsory swinging of this pendulum the monument was coupled in fluctuations and then the fluctuations faded by a natural way. The average value estimation of the logarithmic decrement of the oscillations was equaled 0.055. This figure shows that the metal column with granite shell has rather low capacity to dampen fluctuations. The damper, when it was put in operation during the tests, has increased the ratio of the logarithmic decrement of the oscillations up to the level was equaled 0.18-0.25. The damper construction gives the possibility to obtain greater ratio of logarithmic decrement of the oscillations via increasing of the friction coefficient the energy absorber. It's necessary to note that the spectrum of a structure is its steady characteristic. This function varies with change of mechanical parameters of a structure and can be used for detection of "age" changes of a structure while in exploitation. It's possible to consider that the fixed spectral monument characteristics further can be used as reference for detection of a beginning of the moment "age" changes during a structure-monitoring period.

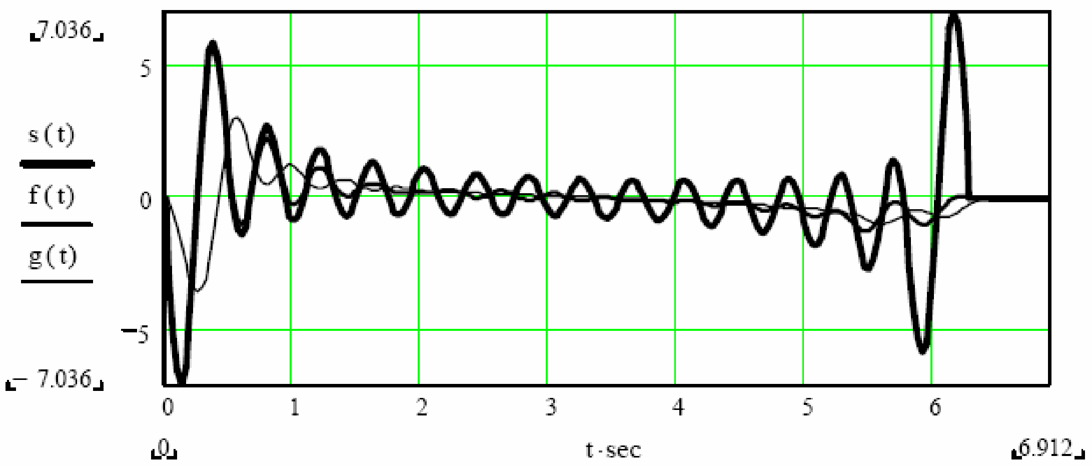


Figure 1

At the figure 1 here are three curves: the first one, which is marked as $s(t)$, is model of sounding signal. The second one ($f(t)$) is misshaped signal by random frequency fluctuation, the third one ($g(t)$) is misshaped signal by random frequency fluctuation and start time fluctuation. Having fulfilled procedure of signal reconstruction one get the curve shape very closed to be the shape of origin one $s(t)$.

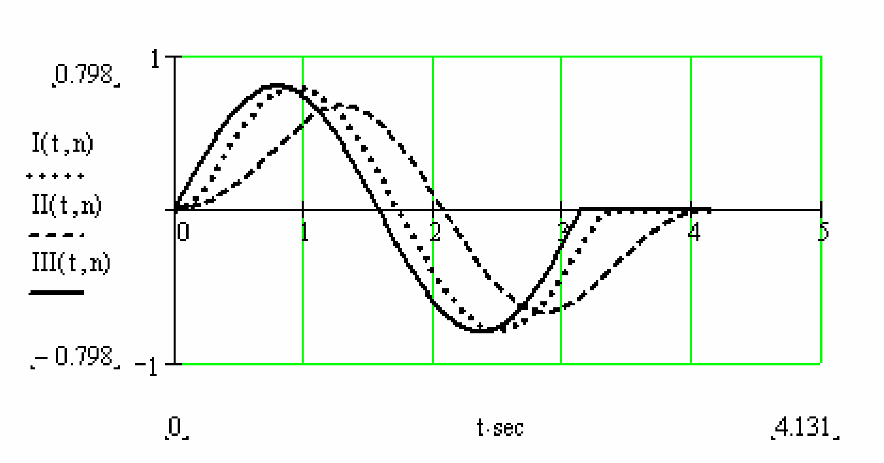


Figure 2

In figure 2 for a case of a signal (13) three curves showing the basic frequency after accumulation procedure are shown. Frequency and interval are functionally connected in the process of fluctuation. Curve I (n) - for a case when fluctuations are symmetric concerning frequency ω_0, II (n) - for a case when fluctuations are not symmetric concerning frequency ω_0 (entropy is equal 2,301 in both cases), III (n) - actually a harmonic without distortions

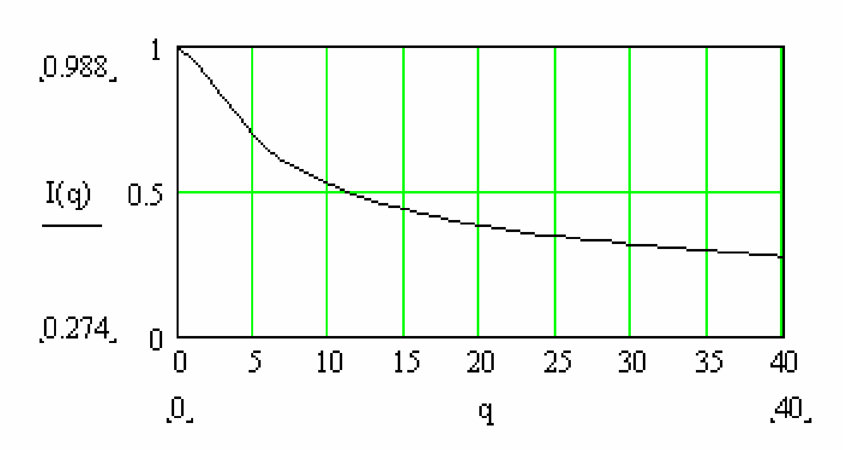


Figure 3

In figure 3 the dependence of a norm of basic function attenuation as a function of number of a harmonic after procedure of accumulation is submitted (in case of the only start moment fluctuations). On an axis of ordinates the amplitude in relative units is shown.

Conclusion

One can find proposed and analyzed original math model of an active monitoring system for manmade and natural objects. The system was used for analyzing of real object characteristics physically. The measurement is based on piling environment response up as a reaction for flow of stochastic weak signals. The response signal correction is used premature probability of instability parameters of testing signals set generator. It is shown that the main source of instability testing signals is not only the time of signal departure but frequency and phase instability as well. For elimination of defects the decision-making procedure is proposed.

Bibliography

1. A.E. Gay, S.V. Mostovoi, V.S. Mostovoi, A.E. Osadchuk. Model and Experimental Studies of the Identification of Oil/Gas Deposits, Using Dynamic Parameters of Active Seismic Monitoring, Geophys. J., 2001, Vol. 20, pp. 895-9009.
2. S. V. Mostovoi, A.E. Gui, V. S. Mostovoi and A. E. Osadchuk Model of Active Structural Monitoring and decision-making for Dynamic Identification of buildings, monuments and engineering facilities. KDS 2003, Varna 2003, p. 97-102
3. Kondra M., Lebedich I., Mostovoi S. Pavlovsky R., Rogozenko V. Modern approaches to assurance of dynamic stability of the pillar type monument with an application of the wind tunnel assisted research and the site measuring of the dynamic characteristics. Eurodyn 2002, Swets & Zeitlinger, Lisse, 2002, p. 1511 - 1515.
4. Mostovoi S., Mostovoi V. et al. Comprehensive aerodynamic and dynamic study of independance of Ukraine monument. Proceadings of the national Aviation University. 2' 2003, pp. 100 - 104.

Authors' Information

Sergey V. Mostovoi – Institute of Geophysics of the National Academy of Sciences, Kiev, Ukraine. e-mail: smost@i.com.ua; most@igph.kiev.ua

Vasiliy S. Mostovoi – Institute of Geophysics of the National Academy of Sciences, Kiev, Ukraine. e-mail: vasmost@i.com.ua; most@igph.kiev.ua

BUILDING DATA WAREHOUSES USING NUMBERED INFORMATION SPACES

Krassimir Markov

Abstract: An approach for organizing the information in the data warehouses is presented in the paper. The possibilities of the numbered information spaces for building data warehouses are discussed. An application is outlined in the paper.

Keywords: Data Warehouses, Operational Data Stores, Numbered Information Spaces

ACM Classification Keywords: E.1 Data structures, E.2 Data storage representations

Introduction

The origin of the Data Warehouses (DW) can be traced to studies at MIT in the 1970s which were targeted at developing an optimal technical architecture [Haisten, 2003]. The initial conception of DW had been proposed by the specialists of IBM using the concept “information warehouses” and its goal was to ensure the access to data stored in no relational systems. In 1988, Barry Devlin and Paul Murphy of IBM Ireland tackled the problem of enterprise integration head-on. They used the term "business data warehouse" and defined it as: “a repository of all required business information” or “the single logical storehouse of all the information used to report on the business” [Devlin and Murphy, 1988]. At present, the conception of “data warehouse” becomes popular mainly due to activity of Bill Inmon. In 1991, he published his first book on data warehousing.

W.H. Inmon's definition is: "Data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision making process" [Inmon, 1991]. Let remember, the data warehouses allow long term information about an enterprise to be recorded, summarized and presented. Usually the data warehouse is a passive observer object that takes no part in business processes, and is not part of the business model. The axes of a multidimensional data warehouse are not arbitrary, but represent real aspects of the business. Axes should represent the purpose, process, resource and organization aspects. The summary hierarchies on each of these axes should parallel the fractal structures in the business model. Roll up and drill down to zoom from summary to detail information is therefore based on the structure of the business, so is meaningful to management and other users. [Marshall, 1997].

As a rule, the typical enterprise has many different systems for operative processing with very incompatible data. In such case, the main task is to convert the existing archives of data into a source for new knowledge which will give to the users a uniform integrated and consolidated notion of the corporate data. The old systems for operative information processing have been developed without foreseeing the support of the requirements of modern business and the need of automated support of decision making. Because of this, the converting the usual systems for online transaction processing (OLTP) in the systems for decision support (resp. – DW) were very complicated task. To solve this problem, an intermediate level has been proposed – the "operational data stores". The **Operational Data Store** (ODS) is a database designed to integrate data from multiple sources to facilitate operations, analysis and reporting. Because the data originates from multiple sources, the integration often involves cleaning, redundancy resolution and business rule enforcement. An ODS is usually designed to contain low level or atomic (indivisible) data such as transactions and prices as opposed to aggregated or summarized data such as net contributions. Aggregated data is usually stored in the DW [Wikipedia, ODS].

The definition of ODS given by Bill Inmon is: "an ODS is a subject-oriented, integrated, volatile, current-valued, detailed-only collection of data in support of an organization's need for up-to-thesecond, operational, integrated, collective information". [Inmon, 1995]

At first glance the ODS appears to be very similar to the data warehouse in structure and content. In some respects there are strong similarities between the two types of architectural constructs. But the ODS has some very different characteristics from the data warehouse. Both the ODS and the data warehouse are subject-oriented and integrated. In that regard, the two environments are identical. Both environments require that data be integrated and transformed as it passes into the ODS and/or the data warehouse. But here the similarities between the ODS and the data warehouse end. The ODS contains volatile data while the data warehouse contains non-volatile data. Data is updated in the ODS while data is not updated in the data warehouse. Another important difference between the two environments is that the ODS contains only very current data while the data warehouse contains both current data and historical data. The data in the data warehouse is not nearly as fresh as the data in the ODS. The data warehouse contains data that is no more current than the last 24 hours. The ODS contains data that may be only seconds old. Another major difference between the two architectural constructs is that the ODS contains detailed data only. The data warehouse contains both detailed and summary data. There are then some major differences between the types of data found in the two environments. One of the most important features of the ODS is the system of record. The system of record is the formal identification of the data in the legacy environment that feeds the ODS. (Figure 1) [Inmon, 1995]

So, an operational data store (ODS) is a type of database often used as an interim area for a data warehouse. Unlike a data warehouse, which contains static data, the contents of the ODS are updated through the course of business operations. An ODS is designed to quickly perform relatively simple queries on small amounts of data (such as finding the status of a customer order), rather than the complex queries on large amounts of data are typical of the data warehouse. An ODS is similar to your short term memory in that it stores only very recent information; in comparison, the data warehouse is more like long term memory in that it stores relatively permanent information.

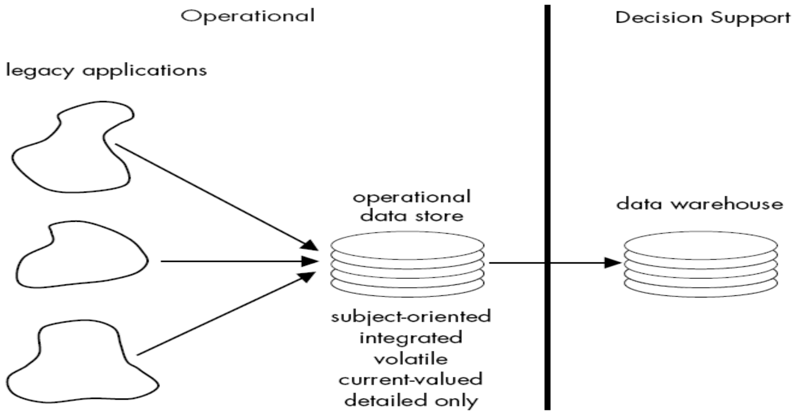


Figure 1. The Operational Data Store [Inmon, 1995]

In the early 1990s, the original ODS systems were developed as a reporting tool for administrative purposes. They were usually updated daily and provided reports about business transactions for that day, such as sales totals or orders filled. This type of system is now referred to as a **Class III ODS**. With changes in technology and business needs, the **Class II ODS** evolved to track more complex information such as product and location codes, and to update the database more frequently (perhaps hourly) to reflect changes. **Class I ODS** systems arose from the development of customer relationship management (CRM). In Class I systems, synchronous or near-synchronous updates are used to provide customers with consistently valid and organized information. Another version, the **Class IV ODS**, was recently developed with an added capacity for more interaction between the data warehouse or data mart and the ODS. [Oracle ODS]

The milestone for the work presented in this paper is the simple idea that we may use a special kind of organization of the information and this way to develop easy to use and compact ODS of Class I with facilities of DW with very high speed for response which enables ***the real-time analytical processing*** (RTAP). (The RTAP multithreaded processing engine needs to support extremely large volumes of data in real time. The analytics performed are composed of combinations of algorithmic, statistical and logical functions. [B-Jensen 2002])

The investigation presented in this paper is based on the fact that a specialized form of data warehouse is the corporate financial ledger. The segments of an account code serve the same purpose as the values on the axes of a data warehouse [Marshall, 1997]. In the same time, there exist a lot of account codes in a financial ledger and it is needed to operate with great complex of tables, descriptions, reports, etc. This leads to very complicated realizations which in the most cases are paid by more and more external memory for hundreds files as well as by growing quantity of processing operations.

In other hand, well-known considerable information complexes are offered by “SAP” (Germany), “Oracle”, “PeopleSoft” (USA), etc., but the prices of such software are very high. This is serious problem for the middle and small enterprises, especially in Bulgaria, which will bankrupt if decide to implement so rich automated systems. Because of this the narrow versions of such software are offered at the market. Unfortunately those versions are not as convenient as they are advertised and provoke many additional problems during the implementation process and exploitation.

Our approach is to build information complexes for information service of business accounting and decision making based on numbered information spaces [Markov, 2004a], which may support RTAP on the level of ODS Class I and this way to reduce the expenses for maintenance separate DW. This goal may be achieved using the FOI Archive Manager (ArM) ®.

FOI Archive Manager (ArM) ®

The FOI Archive Manager (ArM) ® is a tool for building numbered information spaces. ArM is based on the "Multi-Domain Information Model" (MDIM). It has been established more than twenty years ago. For a long period it has been used as a basis for organization of the information bases. The first publication which contains some details from MDIM is [Markov, 1984] but as a whole the model was presented in [Markov, 2004a]. There exist several realizations of FOI Archive Manager (ArM) ® for different hardware and/or software platforms. The newest ArM Version No.:9 for IBM PC developed using DELPHI for MS Windows XP is called **ArM32.**

Let remember the main possibilities of **ArM32** [Markov, 2004b] using some definitions of MDIM.

Basic information element 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. In ArM32 the length of the string may vary from 0 (zero) up to 2^{30} (1G) bytes. There is no limit for the number of strings in an archive but theirs total length plus internal indexes could not exceed 4G bytes in a single file.

Let E_1 is a set of basic information elements: $E_1 = \{e_i \mid e_i \in E_1,\ i=1,\ldots, m_1\}$.

Let μ_1 is a function which defines a biunique correspondence between elements of the set E_1 and elements of the set C_1 of positive integer numbers: $C_1 = \{c_i \mid c_i \in N,\ i:=1,\ldots, m_1\}$, i.e. $\mu_1 : E_1 \leftrightarrow C_1$. The elements of C_1 are said to be number codes of the elements of E_1. The triple $\mathcal{S}_1 = (E_1 , \mu_1 , C_1)$ is said to be a *numbered information space of range 1*.

The triple $\mathcal{S}_2 = (\mathcal{E}_2, \mu_2, C_2)$ is said to be a *numbered information space of range 2* iff $\mathcal{E}_2$ is a set which elements are numbered information spaces of range 1 and μ_2 is a function which defines a biunique correspondence between elements of $\mathcal{E}_2$ and elements of the set C_2 of positive integer numbers: $C_2 = \{c_j \mid c_j \in N,\ j:=1,\ldots,m_2\}$, i.e. $\mu_2 : \mathcal{E}_2 \leftrightarrow C_2$.

The triple $\mathcal{S}_n = (\mathcal{E}_n, \mu_n, C_n)$ is said to be a *numbered information space of range n* iff $\mathcal{E}_n$ is a set which elements are information spaces of range *n-1* and μ_n is a function which defines a biunique correspondence between elements of $\mathcal{E}_n$ and elements of the set C_n of positive integer numbers: $C_n = \{c_k \mid c_j \in N,\ k:=1,\ldots,m_n\}$, i.e. $\mu_n : \mathcal{E}_n \leftrightarrow C_n$.

The sequence $A = (c_n,c_{n-1},\ldots,c_1)$ where $c_i \in C_i$, i=1,…,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 $A_1,A_2,\ldots,A_k$, where k is arbitrary positive number, is said to be a *space index*.

Every index may be considered as 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. So, every index which points only to indexes is called *metaindex*.

Let G = {S_i | i=1,...,m} is a set of numbered information spaces.

Let τ={ν_{ij} : $S_i \to S_j$ | i=*const*, j=1,…m} is a set of mappings of one “main” numbered information space $S_i \subset G$, i=*const*, into the others $S_j \subset G$, j=1,…m, and, in particular, into itself. The couple: *Đ* = (G, τ) is said to be an “*aggregate*”.

The **ArM32** elements are organized in numbered information spaces with variable ranges. There is no limit for the ranges the spaces. Every element may be accessed by correspond multidimensional space address (coordinates) given via coordinate array of type cardinal. At the first place of this array the space range needs to be given. So, we have two main constructs of the physical organizations of ArM32 – numbered information spaces and elements.

The main **ArM32** 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 **ArM32** numbered information spaces are ordered and main operations within spaces take in account this order. So, from given space point (element or subspace) we may search the previous or next empty or non empty point (element or subspace). In is convenient to have operation for deleting the space as well as for count its nonempty elements or subspaces.

The **ArM32** logical operations defined in the multi-domain information model are based on the classical logical operations - intersection, union and supplement, but these operations are not so trivial. Because of complexity of the structure of the spaces these operations have at least two principally different realizations based on codes of information spaces’ elements and on contents of those elements.

The **ArM32** information operations can be grouped into four sets corresponding to the main information structures: elements, spaces, aggregates, and indexes. Information operations are context depended and need special realizations for concrete purposes. Such well known operations are, for instance, transferring from one structure to another, information search, sorting, making reports, etc.

At the end there exist several operations which serve information exchange between **ArM32** archives (files) such as copying and moving spaces from one to another archive.

ArM32 engine supports multithreaded concurrent access to the information base in real time.

Very important feature of ArM32 is possibility not to occupy disk space for empty structures (elements or spaces). Really, only non empty structures need to be saved on external memory.

Complex FOI®

Complex FOI® is an integrated software environment for economical information processing and business analysis. The main features of **Complex FOI** [Markov et al, 1994] are built on three levels, which correspond to the Pyramidal Information Model (PIM) presented in [Markov et al, 1993]. The levels of this model are “Strategy”, “Analysis”, and “Service”. Every level contains three parts, which correspond to “Human Resources”, “Materials”, and “Finances” of the enterprise. It easy to see that there exist correspondence between PIM and ODS and DW.

The main set of concrete systems for information processing is included on "Service" level. They are aimed to service the operative work and control. For instance, there exist systems for service the enterprise financial tasks such as computing of salaries [Markov et al, 1996a], systems for managing different material stores using appropriate information access - by names or by numbers of goods [Markov et al, 1995a], systems for maintenance of fixed assets [Markov et al, 1996b], etc. An example of another class of service systems is one for automated payment of consumption of water and other communal services in a town as well as the specialized service systems, such as one for computing the price of building of some architectural object. It is clear, *the legacy applications* of the enterprise are assumed to be on this level too.

All these systems are integrated with the upper level ("Analysis") via very convenient interface – the natural language standard accounting records which are the usual transaction form for accounting process. Furthermore, the information in **Complex FOI** is distributed in correspond numbered information spaces in accordance to usual every day financial accounting information structures. This make integration possible and automated information exchange is simple and comprehensible.

There is only one system on level "Analysis". It is an ODS with possibilities for accounting as well as for account analysis [Markov et al, 1995b]. This is the main tool for enterprise financial control and managing which support automated day-to-day operations (purchasing, banking etc), transactions access and modifying a few records at a time, application oriented database design, and metric: transactions/sec. The main structure of this level is the financial ledger - usually it is a numbered information space of range up to 10. Its subspaces represent accounting divisions, groups and accounts, as well as sub-accounts on several sub-levels. Every space may contain operational and historical data in the same time.

The main feature of the level "Strategy" is the decision support. All information from low levels can be used for supporting the processes of business decisions in the group of leaders of the enterprise. The functionality of this level covers the usual understanding of data warehouse but it is realized as distributed RTAP engine which support complex queries that access records with operational and/or historical data for trend analysis.

Because of special multidimensional organization, in Complex FOI the analytical pre-computation can be provided in real time during the operative work and its results (elements, spaces, aggregates, and indexes) can be stored in corresponded structures of the multidimensional hierarchical information base. So, in query response time, it is easy to process *multidimensional modeling* (for instance - compute total *sales* volume per *product* and *store*); *operating with dimensions and hierarchies* (for instance - roll-up: move up the hierarchy e.g. given total salaries per department, we can roll-up to get salaries per enterprise; drill-down: move down the hierarchy more fine-grained aggregation; pivoting: aggregate on selected dimensions usually 2 dims (cross-tabulation)); *comparisons* (for instance - this period vs. last period - show me the sales per store for this year and compare it to that of the previous year to identify discrepancies); *ranking and statistical profiles* (for instance – top N / bottom N - show me sales, profit and average call volume per day for my 10 most profitable salespeople); *custom consolidation* (for instance - market segments, ad hoc groups - show me an abbreviated income statement by quarter for the last four quarters for my northeast region operations); etc.

Conclusion

The approach to build information complexes for information service of business accounting and decision making based on numbered information spaces which may support RTAP on the level of ODS Class I and this way to reduce the expenses for maintenance separate DW has been presented in the paper. This goal may be achieved using the FOI Archive Manager (ArM) ® and "Multi-Domain Information Model" (MDIM). An application of presented approach named "Complex FOI" was outlined.

Acknowledgments

Author is indebted to Ilia Mitov and Krassimira Ivanova for support and collaboration. Due to theirs hard work the approach presented in this paper has been widely implemented in practice.

This work is a part of the project "**ITHEA XXI**", partially financed by the Consortium FOI Bulgaria.

Bibliography

[B-Jensen 2002] M.T. B-Jensen. High Tower Software's Tower View is the Odds-On Favorite of International Game Technology for Real-Time Data Management. Product Review published in DM Review Magazine July 2002 Issue. http://www.dmreview.com/article_sub.cfm?articleId=5403

[Devlin and Murphy, 1988] B.A. Devlin and P.T. Murphy. An Architecture for a Business and Information System. IBM Systems Journal. Volume 27, No. 1, 1988. http://www.research.ibm.com/journal/sj/271/ibmsj2701G.pdf

[Haisten, 2003] M. Haisten. The Real-Time Data Warehouse: The Next Stage in Data Warehouse Evolution http://www.damanconsulting.com/company/articles/dwrealtime.htm

[Inmon, 1991] W.H. Inmon. Building the Data Warehouse, QED/Wiley, 1991.

[Inmon, 1995] W.H. Inmon. The Operational Data Store. InfoDB February 1995 http://www.evaltech.com/wpapers/ODS2.pdf

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

[Markov et al, 1993] K. Markov, K. Ivanova, I. Mitov, J. Ikonomov. Pyramidal Model of the Firm Information Activities. IJ ITA, 1993, Vol. 1, No. 2. (in Russian)

[Markov et al, 1994] K. Markov, K. Ivanova, I. Mitov. Basic concepts and main information structures of the Complex FOI. FOI-COMMERCE, Sofia, 1994. (in Bulgarian)

[Markov et al, 1995a] K. Markov, K. Ivanova, I. Mitov. Automated service of the storehouses. FOI-COMMERCE, Sofia, 1995. (in Bulgarian)

[Markov et al, 1995b] K. Markov, K. Ivanova, I. Mitov. Automated service of the financial accounting using System "ANALYSE". FOI-COMMERCE, Sofia, 1995. (in Bulgarian)

[Markov et al, 1996a] K. Markov, K. Ivanova, I. Mitov. Automated service of the accounting the staff and salaries FOI-COMMERCE, Sofia, 1996. (in Bulgarian)

[Markov et al, 1996b] K. Markov, K. Ivanova, I. Mitov. Automated service of the accounting of the fixed assets. FOI-COMMERCE, Sofia, 1996. (in Bulgarian)

[Markov, 2004a] K. Markov*. Multi-Domain Information Model.* Proceedings of the ITC&P-2004 - International Conference "Information Technologies and Communications & Programming", Varna. FOI-COMMERCE, 2004, pp. 79-88. Int. Journal "Information Theories and Applications", 2004, Vol. 11, No. 4, pp. 303-308

[Markov, 2004b] K. Markov. *Coordinate Based Physical Organization of Computer Representation of Information Spaces*. Proceedings of the Second International Conference "Information Research, Applications and Education" i.TECH 2004, Varna, Bulgaria. Sofia, FOI-COMMERCE – 2004, стр.163-172 (in Bulgarian).

[Marshall, 1997] Cr. Marshall. *Business Object Management Architecture*. OOPSLA'96 Workshop Business Object Design and Implementation II: Business Objects as Distributed Application Components - the enterprise solution? http://jeffsutherland.com/oopsla97/marshall.html

[Oracle ODS] - whatis.com - http://searchoracle.techtarget.com/sDefinition/0,,sid41_gci786730,00.htm

[Wikipedia, ODS] http://en.wikipedia.org/wiki/Operational_data_store

Authors' Information

Krassimir Markov – Institute of Mathematics and Informatics, BAS, e-mail: foi@nlcv.net

TABLE OF CONTENTS OF VOLUME 12, NUMBER 2

Neural Control of Chaos and Aplications 103
Cristina Hernández, Juan Castellanos, Rafael Gonzalo, Valentín Palencia

Bridging the Gap Between Human Language and Computer-Oriented Representations 111
Jesús Cardeñosa, Carolina Gallardo, Eugenio Santos

A New Approach for Eliminating the Spurious States in Recurrent Neural Networks 119
Víctor Giménez-Martínez, Carmen Torres, José Joaquín Erviti Anaut, Mercedes Perez-Castellanos

A Geometrical Interpretation to Define Contradiction Degrees between Two Fuzzy Sets 127
Carmen Torres, Elena Castiñeira, Susana Cubillo, Victoria Zarzosa

A Workbench for Document Processing 135
Karola Witschurke

Experiments in Detection and Correction of Russian Malapropisms by Means of the WEB 141
Elena Bolshakova, Igor Bolshakov, Alexey Kotlyarov

A Mathematical Apparatus for Domain Ontology Simulation. An Extendable Language of Applied Logic 149
Alexander Kleshchev, Irene Artemjeva

Information Processing in a Cognitive Model of NLP 157
Velina Slavova, Alona Soschen, Luke Immes

Mathematical Model of Re-Structuring Complex Technical and Economic Structures 166
May Kornijchuk, Inna Sovtus, Eugeny Tsaregradskyy

DIAGaRa: An Incremental Algorithm for Inferring Implicative Rules from Examples 171
Xenia Naidenova

Active Monitoring and Decision Making Problem 186
Sergey Mostovoi, Vasiliy Mostovoi

Building Data Warehouses Using Numbered Information Spaces 193
Krassimir Markov

Table of Contents of Volume 12, Number 2 200