ITHEA

International Journal INFORMATION THEORIES & APPLICATIONS

2017 Volume 24 Number 1

International Journal

INFORMATION THEORIES & APPLICATIONS
Volume 24 / 2017, Number 1

International Journal “INFORMATION THEORIES & APPLICATIONS” (IJ ITA)
is official publisher of the scientific papers of the members of
the ITHEA International Scientific Society

IJ ITA welcomes scientific papers connected with any information theory or its application.
IJ ITA rules for preparing the manuscripts are compulsory.
The **rules for the papers** for IJ ITA are given on *www.ithea.org*.

Responsibility for papers *published in* IJ ITA belongs to authors.

International Journal “INFORMATION THEORIES & APPLICATIONS” Vol. 24, Number 1, 2017

Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria, in collaboration with:
Institute of Mathematics and Informatics, BAS, Bulgaria,
V.M.Glushkov Institute of Cybernetics of NAS, Ukraine,
Universidad Politécnica de Madrid, Spain,
Hasselt University, Belgium,
St. Petersburg Institute of Informatics, RAS, Russia,
Institute for Informatics and Automation Problems, NAS of the Republic of Armenia

Printed in Bulgaria
Publisher ITHEA®
Sofia, 1000, P.O.B. 775, Bulgaria. www.ithea.org, e-mail: office@ithea.org
Technical editor: **Ina Markova**

ISSN 1310-0513 (printed) **ISSN 1313-0463 (online)**

UNRAVELING BI-LINGUAL MULTI-FEATURE BASED TEXT CLASSIFICATION: A CASE STUDY

Aziz Yarahmadi, Mathijs Creemers, Hamzah Qabbaah, Koen Vanhoof

Abstract: *Extracting knowledge out of unstructured text has attracted many experts in both academia and business sectors like media, logistics, telecommunication and production. In this context, classification techniques are increasing the potential of Natural Language Processing in order to produce an efficient application of text classification in business context. This method could extract patterns from desirable text. The main objective of this paper is implementing a classification system which can be widely applied in commercial product classification problem solving. We have employed various applications of Natural Language Processing and Data Mining in order to solve parcel classification problem. Furthermore, we have investigated a popular case study which is associated with parcel shipping companies all around the world. The proposed methodology in this paper is part of a supervised machine learning project undertaken in order to gain domain specific knowledge from text.*

Keywords: *supervised text mining; commodity description classification; shipment classification system; natural language processing*

1. Introduction

Supervised machine learning frameworks have been introduced to business decision making in several domains in order to simplify tremendous complex datasets which represent few or no guidelines in data interpretation. On the other hand, more and more organizations are interested to apply such frameworks to documents in which lies patterns that can be modified towards growth and clarification. Text mining, which intensifies the hidden structure in documents, is recently well integrated in classification functions. Feldman and Sanger [Feldman, Sanger, 2006] express the similarities between Text Mining and Data mining due to the fact that text mining is originated from data mining and inherits much of data mining techniques in the text processing context.

Recent research has focused on applications related to text pattern recognition in business planning and operations. Current examples are deepened in sentiment analysis applications in marketing, social media mining for product and service penetration, fraud detection in financial statements, customer service and many more. Unique potentials of pattern discovery in documents, from structured to totally unstructured, empowers business administrators to extract rules even from informal and manipulated sentences. However, applications in this context are sensitive to the text structure and vary on the use and the expected results. An attractive case study for companies producing and transferring commercial goods is to employ a classification system for the items they produce, send or receive. Such a system could be highly valuable to inbound and outbound logistic services as well [Shankar, Lin, 2016]. The combination of text mining and supervised machine learning is known as corpus-based document classification. Furthermore, in this context, documents are being annotated manually in order to create corpora, that acts as a relational database of documents, categorized and supervised to be used later on, to assess and analyze consecutive document datasets.

Commercial commodity categorization using text mining implements a classification framework using pre-labeled items to train a classifier [Kotsiantis, 2007]. Moreover, the ontology behind supervised text classification highlights the need for enough data to train the classifier. In other words, problem solving cases dealing with commodity classification focuses on using commodity features to group similar items. Consequently, the research topic in this paper focuses on forming a corpus of documents with pre-determined structure required to form the classification model. Additionally, we present a methodology to address the question of classifying commodity descriptive features by dealing with parcel delivery records with commodity description in two languages: English and Arabic. We are proposing a three step framework to analyze and categorize products in 70 commodity description groups provided by the parcel delivery company:

1. Data cleansing, pre-processing and translation
2. Corpus creation and class labeling
3. Model selection and results verification

The main objectives of our study is to classify items in big datasets of delivered commodity descriptions including multiple languages and to normalize categories into a set of pre-defined items in company's interest. Here, the second main approach in text mining which uses pre-defined dictionary of terms would also seem reasonable to perform. We will also illustrate the implementation of "Wordnet", as the widely used lexicon-or dictionary, to categorize items and the shortcomings of such method in this case study. The rest of this paper is organized as follows. Section 2 will go through recent study and evidence applying text mining rules in text classification. Section 3 focuses on the proposed framework based on the structure of data in this study. In section 4, model selection is discussed in details. The results of the proposed framework have been displayed in section 5. Finally, section 6 outlines the conclusions.

2. Recent work

The state of the art in text mining mainly focuses on sentiment analysis. The goal in sentiment analysis is to extract the modality of user behavior expressed in comments towards a subject. Applications tend to classify the reviews in bipolar classes of negative or positive. Moreover, commodity classification has been increasingly explored from the quantity point of view which reports on the number of entities instead of classes themselves. Commodity description features refine the classification task in this study though. Underlying qualitative features like the use case, nature and the customer segment, categorize the commodity description.

Ghani et al [Ghani et al, 2006], have represented the idea of product identification based on series of attribute-value pairs. Thus, such a methodology allows feature vector extraction that supports the term frequency framework in text classification.

Popescu and Etzioni [Popescu, Etzioni, 2005] have built the model called OPINE based on the review scores and features extracted from customer reviews. A classification framework based on these features can be constructed. Furthermore, feature extraction in text classification is of great importance. Although different definitions of a feature might be found due to context, a feature in text mining is theoretically the presence or absence of each word in text. Distinctive features are to be scored higher in tf-idf method, a significant feature for grouping similar documents.

The concepts for tf-idf are separately discussed in [Luhn, 1957] and [Spärck Jones, 1957]. Recently the application of "Wordnet" in lexicon-based text classification has created opportunities in semantic mappings in product classification [Beneventano et al, 2004]. Although lexicon-based text mining approaches employ dictionaries of word trees, with broad synsets, which are synonyms for English words, a supervised classification approach determines manually categorized descriptions for item classes, which would result in higher precision and recall degrees for the classification model. On the other hand, the use of dictionaries on item class description requires clear description words, which in this case is limited and no specific study has undertaken such an approach so far.

3. Proposed methodology

Representation of the case study in this research focuses on the application of current text classification techniques on a common data classification problem in business context. Due to lack of matching research applications, we have been motivated to answer commercial production request for more information regarding classification of products based on their technical features, use cases, age and gender specifications. Customer segmentation and profiling are two major fields of study by categorizing parcel packages. Parcel delivery companies are not normally able to open packages to get insights into the contents. Commodity features are provided by the merchants that act as package senders. These types of information are represented using various languages. On the contrary to opinion mining problems dealing with the polarity of features dependent to an entity, the use of dictionaries is limited to text classification with few text features. In this perspective, we propose a supervised classification methodology for commodity classification based on commodity descriptions. In order to unravel the commodity classification case study, a data structure deliberation is presented and the need for our research methodology is introduced.

3.1 Data

Commodity classification task we have at hand deals with a dataset of hundred thousand items of commodities delivered to customers by a parcel delivery company. The parcel delivery company has invested in two major delivery systems:

The first one is an international e-commerce based shopping ship service that aims to connect major merchandising and shopping platforms like eBay and Amazon with potential customers in countries with less product delivery coverage. 250,000 customers have recently used this initiative.

The second service offered by the company is their domestic parcel delivery system that is generating loads of data regarding package features. Statistics showed that 3 million users have used this service. The parcel delivery company has focused on three main merchants with the highest market shares.

The original dataset forms the parcel delivery record including 56 parcel features. The feature types vary from size dimensions and weight to payment type and destination. The commodity description feature is very broad and the need for a general categorization into 70 pre-defined classes is inevitable by the parcel delivery company. Assigning a class to each commodity becomes challenging when the only feature to be used for classification is the commodity description as it provides information about the name, quantity and brand of the product delivered to the customer. However, classification based on size dimensions is not in the investigation of this study, considering that there is no record of classification based on the dimensions of the commodity. Still, description features provide a range of descriptive properties for each item that introduce the brand, technical specifications, model, etc. As commodity description shares various property names with the original classes, it is reasonable to take it into consideration as the feature extraction source.

The challenge arises when the description feature contains non-ascii characters (Table 1). Text preprocessing is to be done in order to normalize the text.

Commodity Description
#1/DKNY WOMEN WATCH MODEL NO NY8833
#1/براون سيلك ابيل لازاله الشعر و تحديد الحواجب BRAUN
#1/Aramis Brown for Men 110ml Eau de Toilette
#1/RGB Led Strip Waterproof 5M SMD 5050 300
#1/كيراتين مرطب لا يشطف 177مل
#1/Braun SE 5780 Silk Epil 5 Epilator

Table 1: a sample of data before preprocessing

3.2 Preprocessing advantages

The main obstacle in normalizing text in this dataset is the fact that in some cases, the commodity description is added in Arabic or it is a combination of Arabic and English. Such text inputs would drastically lower the classifier accuracy as wrong or no specific classes would be assigned to commodities with such descriptions. While dealing with big datasets, one approach could be to remove commodity records with descriptions in Arabic. Our text classification approach however, proposes to translate the descriptions into English. Translation function accuracy is assessed by native Arabic speakers in order to remove the outliers and less precise translations.

Furthermore, in some cases, description feature states names which represent distinctive classes. Except for size and weight, that might be able to provide more distinctive feature for each commodity description in these cases, it is not possible to figure out the actual class. As the classification is built on description only, these records are to be removed from the training and test set. Removing punctuation marks and English stop words are respectively the next steps in this dataset, as the focus in item classification is on proper nouns. Numbers and hash tags are as well removed as they will not create added value. Missing values from commodity description column are few but possible. Removing such records is necessary along with duplicate items. Further, we implemented stemming in order to normalize further proper commodity names. The role of stemmers in text preprocessing is to reduce the number of derived forms of words. Removing prefixes is the other advantage in this case. For large corpuses, stemmers have led to better results [Vijayarani et al, 2001]. In our case, all forms of “package”, “packages” and “packaging” are relatives of the basic form “package”. Similarly, stemming significantly impacts the multiple forms of verbs usage. Figure1 is visualizing the schematic of preprocessing steps which were applied in this study. In the next part, we will go through model selection.

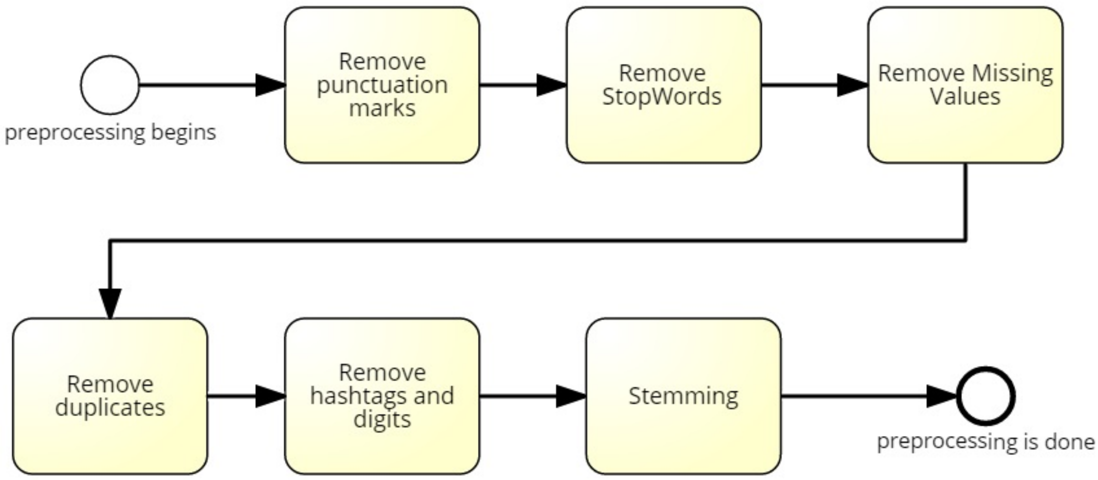


Figure 1: preprocessing steps implemented using Signavio Process Editor

4. Model selection based on text structure

Natural language processing tasks vary from part of text tokenization to sentiment detection and therefore, multiple probability distribution estimation techniques can be applied to text classification projects. Our desired task is to assign a class to a set of words using pre-labeled training data. 70 pre-defined classes imply the use of supervised text classification. The algorithm used to assign respective classes requires a set of features for each item as training data. A feature is primarily a conversion function to transform each input value to a set of input values and relative label given to it. The machine learning algorithm then models these feature sets and extracts a classification schematic that can be used for incoming unlabeled data. The transformation function will be implemented on new data and feature extraction is performed again and the predicted model will output results, depending on the classification algorithm used [Aggarwal et al, 2012].

Since classifying our feature description can be viewed from the unsupervised text classification point of view as well, the most noted lexicon, or dictionary to be used in such a methodology is "Wordnet". Recent use of Wordnet has spread in academic text classification use cases due to comprehensive network of word trees, creating a . Initial use of Wordnet was a step toward building language model using treebanks of synonyms and hyponyms of words. The NLTK package for Python represents a sound implementation of Wordnet and has gathered tools to perform various NLP information extraction techniques. Despite huge capabilities of using it, the obstacle on our way is linked with the pre-defined 70 restricted classes. Our implementation of Wordnet was based on evaluating the commodity description word by word. For each word, we found all synonyms. Lookup function would then match if any of the words would match any synonyms found in Wordnet. If not, we moved one level deeper in the word tree and tried again. Such a method would try the same procedure on bi-grams on the next phase and would try the N-gram model for N=3 and N=4 as well. While using Wordnet, which is basically a dictionary of each word and its categorical mappings, is a type of unsupervised machine learning, it has proved to be accurate on document classification. Restricted classes we have in our study did not map correctly and precisely onto equivalent Wordnet categories, and that is why using supervised corpus based text classification is preferred in this study

Choosing the right model has been studied in this research among a set of classification algorithms. A comparative statistical analysis will show the state-of-the-art modeling capabilities of each classifier. Two popular classification algorithms in this context are Naïve Bayes and K-Nearest Neighbor, due to their predictive capabilities and precise performance. Naïve Bayes will be error-prone for small datasets while it is easy to implement in text categorization problems. However, with regard to classification accuracy it will not be preferred over Support Vector Machines [Rennie et al, 2003] [Zhang, Oles, 2011]. While Support Vector Machines (SVMs) has been broadly accepted for text classification tasks [Joachims, 1998] [Pang et al, 2002], we have developed models based on underrated classification algorithms. These include Boosting, GLMNET, MAXENT and SLDA.

[Kudo, Matsumoto, 2004] has introduced a framework based on Boosting algorithms to classify semi-structured sentences represented as a labeled ordered tree. The paper presents the subtrees as Features extracted from text. Boosting algorithms leverage the weak learners, classifiers which predict the right class only slightly better than random guessing, to become strong learners, which would correlate rigorously with the right classification. Each classifier will be trained based on the hardest instances to classify by previous classifier [Schapire, Singer, 2000]. Implementations for GLMNET, MAXENT and supervised LDA are available using R packages. MAXENT algorithm in our implementations has achieved the highest precision and recall measures. Maximum Entropy classifier (abbreviated in MAXENT), has proved the efficiency of the classifier in text categorization scope.

Della Pietra et al. [Della Pietra et al, 1997] state that given a set of classes C with

$$C : \{c_1, c_2, c_3, \dots c_N\}$$

A set of labeled documents with classes:

$$(d_1, c_1), (d_2, c_2), \dots (d_n, c_N)$$

For the document d and the pertinent class c, MAXENT representation of $P(c|d)$ is an exponential form of:

$$P(c|d) := \frac{1}{Z(d)}\, exp\, (\sum_{i} \lambda_{i,c}\, F_{i,c}\, (d, c))$$

here, $\lambda_{i,c}$ is the set of features and their weights. $F_{i,c}$ is the feature in this model and $Z(d)$ is the normalization function:

$$Z(d) = \sum_{c} exp\, (\sum_{i} \lambda_{i,c}\, F_{i,c}\, (d, c))$$

$F_{i,c}$ is a binary function and depending the relativity of the feature selection to context, the output for each instance to be classified is either zero or one [Chieu, Ng, 2002]. Theoretically speaking, features in Maximum Entropy model are a function of classes predicted and a binary context constraints. [Ratnaparkhi, 1996] has introduced the representation of such a function in his part-of-speech tagger. A feature in this explanation is a set of yes/no questions with a constraint applied to each word. Any word that satisfies the constraint would be a feature. Maximum entropy models state the fact that among all probability distributions available to model testable data, which in this study is the groupings of the commodities, the one model with the highest entropy is the true model. The main substantial feature in maximum entropy in this study is the ability to handle comprehensive features which will be the case here. Document term matrix or feature matrix is created using the package tm of R, and it is the nominal representation of the most distinctive terms used in each class for categorization. Additionally, a feature-cutoff will leave feature seen than a specific number of times. Figure 2 represents an overview of model generation procedure in this study.

We have used the RTextTools library for machine learning in text classification tasks. The R distribution of MAXENT, trains a maximum entropy model using a document term matrix and feature vector. Document term matrices are numerical representations of documents. Each document would be represented word by word in rows and the columns state the existence or absence of words in each document. A feature vector would as well represent the given labels. Consequently, the trained model will be tested with new unlabeled data. To train models, there are labelled datasets from headlines of New York Times and bills from United States Congress.

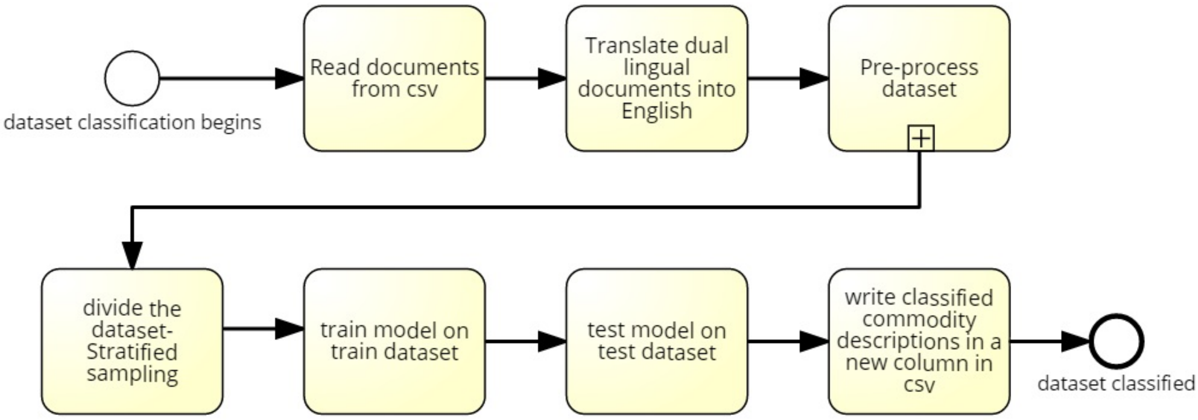


Figure 2: Model generation steps implemented using Signavio Process Editor

5. Discussion and Results

Using stratified sampling and with respect to 70 classes at hand, we have assigned %80 of data to training data and the model trained on this portion will be applied to the rest %20. Data loaded will be used to create a document-term matrix. Preprocessing options are available in this step. In the next step a container will be created from the document-term matrix that acts as a list of objects for the machine learning algorithm. Training models will be initiated next and classified data based on the trained models will be provided consequently. Finally, the analytics for classification task will be provided and the results will be exported to the output file desired by the user [Jurka et al, 2013].

Applying our classification methodology to new unlabeled data consistently provided by our parcel delivery partner proved to be highly efficient both theoretically and empirically. The classification model achieved the precision and recall measures of 0.9365217 and 0.9144928, respectively. Precision measure introduces the number of right returned instances that are queried by the classifier.

$$Precision = \frac{true\ positive(t_p)}{true\ positive(t_p) + false\ positive(f_p)}$$

In this equation, the true positives and false positives are respectively the number of hits or true instances to be found and the number of instances that are selected as hit but belong to other groups. In other words, precision detects the percentage of relevant items selected. Recall is the other performance measure for classification tasks and it detects the percentage of selected items that are relevant.

$$Recall = \frac{true\ prositive(t_p)}{true\ prositive(t_p) + false\ negative(f_n)}$$

False negative in this equation indicate the case where an instance is rejected to be in the respective class, while it actually belongs to the class. Precision and recall are both indications of the relevancy of model. Table 2 shows precision and recall rates for the five classification algorithms applied on our dataset. As the table represents, Support Vector Machine algorithm shows lower precision and recall rates, compared to Max Entropy and the reason lies in the fact that text data is not a complete relevant input for SVM. Vector representation of text results in sparse matrices and these matrices will not lead to the highest ranked results in SVM.

Algorithm	**Precision**	**Recall**	**F-score**
GLMNET	0.5207246	0.4208696	0.4407246
MAXENT	0.9365217	0.9144928	0.9228986
BOOSTING	0.9502899	0.9165217	0.9260870
SLDA	0.8911594	0.8952174	0.8839130
SVM	0.9026087	0.8834783	0.8882609

Table 2: precision, recall and F.score for each model

Some of the models that have been tested in this study have showed close precision and recall measures. With respect to precision, Boosting, MAXENT, SVM and SLDA have gained the highest ranks respectively, outperforming GLMNET significantly with the precision roughly around %52. Recall measure for GLMNET is low as well compared to the rest of the models. Big number of classes in our case study lowers the results for GLMNENT classifier. Furthermore, GLMNET is a low-memory classifier and the size of case study dataset is much bigger than the 30,000 text documents it can handle. MAXENT and BOOSTING gain very close recall measures, followed by SLDA which would outperform SVM by around %1. Figure 3 and figure 4 represent the visualization for precision and recall achievements for all models tested in this study.

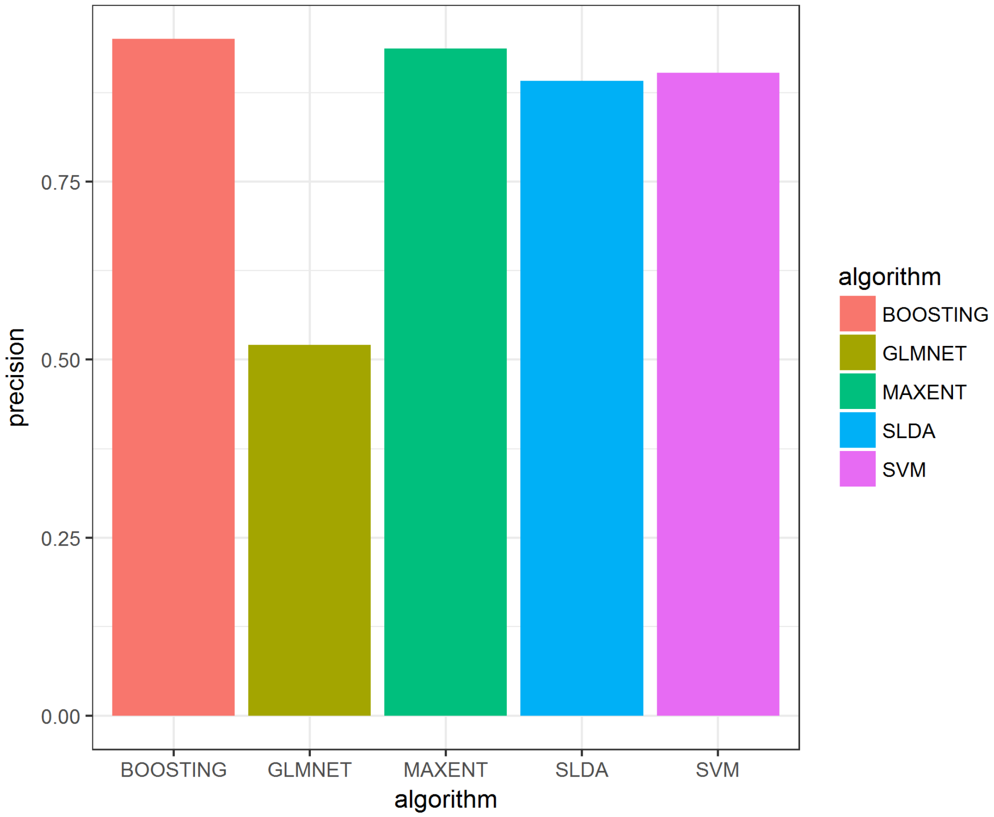


Figure 3: precision comparison charts for all models

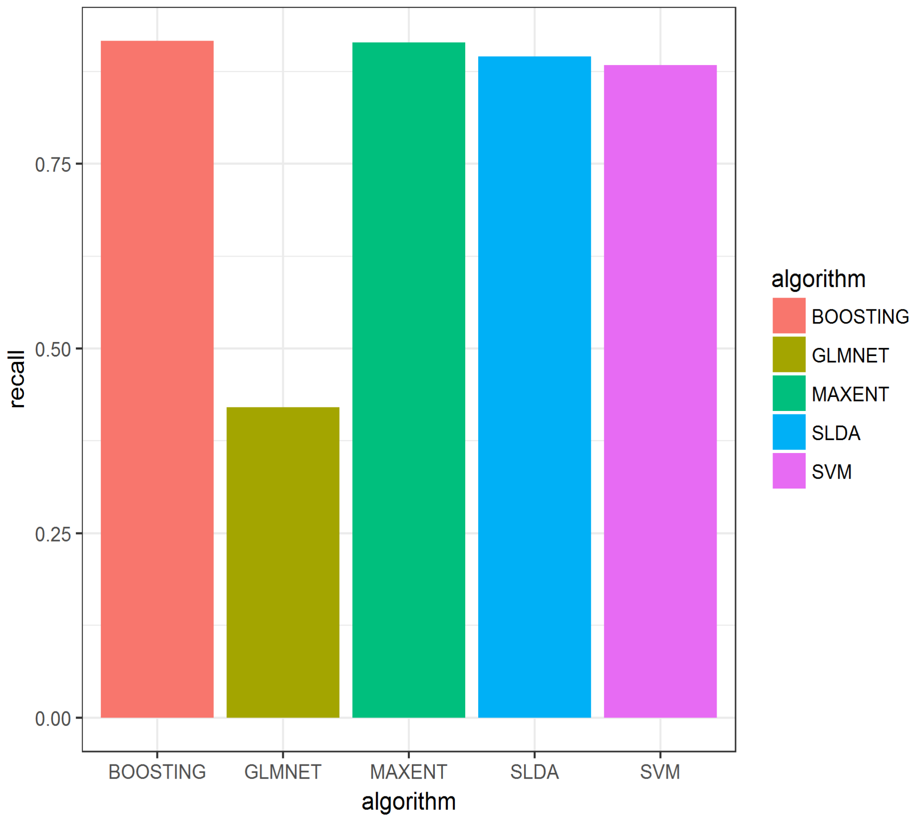


Figure 4: recall comparison charts for all models

The F-score for each classification algorithm is illustrated in table 2. The F-score is simply the harmonic mean of precision and recall. For the case where precision and recall are both one, we multiple the measure by 2.

$$F = \frac{precision * recall}{precision + recall}$$

The highest F-score is achieved by BOOSTING and MAXENT respectively, followed by SV and as expected, the F-score for GLMNET is considerably lower, around %44. Figure 5 represents the comparison of F-score achieved by all models in this study.

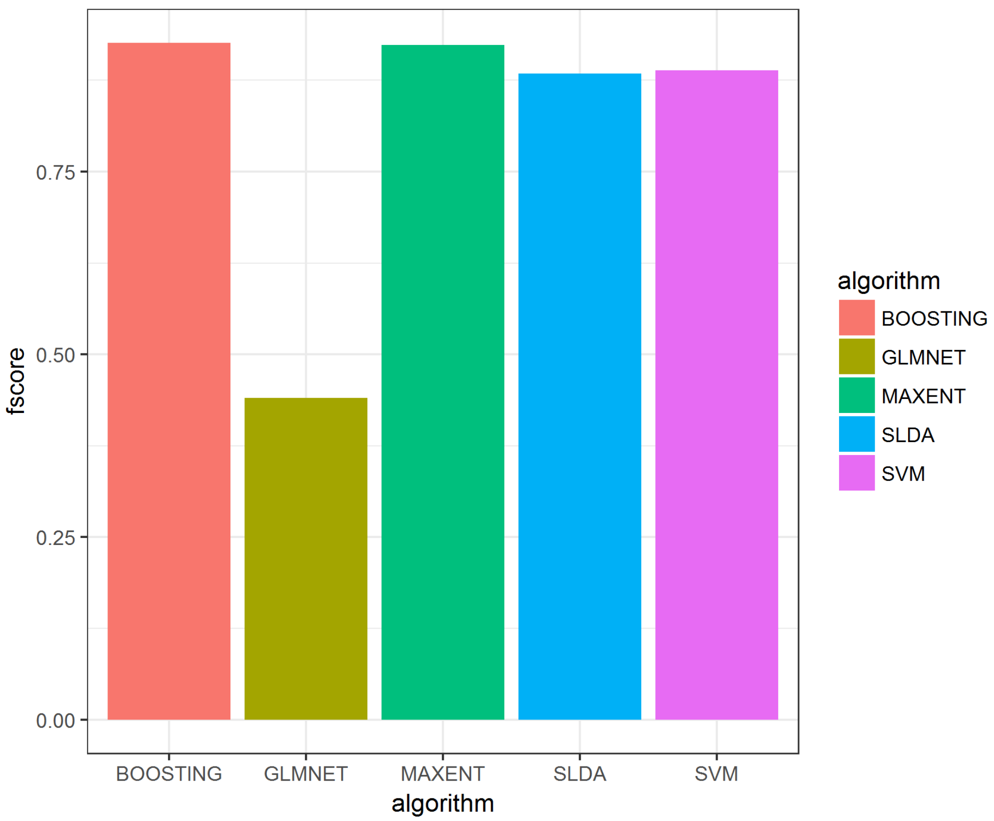


Figure 5: F-score comparison results for all models

Our implementation in R would read the unlabeled data from a csv file and the output would be written to a new column on the csv file, making it easy to transform complicated descriptions into a definite set of 70 classes. Low confidence labels are put in separate category which is called "Others". These instances are not to receive a class tag or have loose bonds with any of the classes. In other words, the feature set extracted from these description items is by no means clearly showing pointing to any of the 70 classes defined.

The implementation of MAXENT classifier in R has proved to be memory friendly. Models for MAXENT and GLMNET can run on local hosts while the rest of the models can run out of memory even on our access point to Flemish Supercomputer that is a powerful computing node, which in fact originates from the vast span features and high dimensional vector matrices.

6. Conclusion and future work

Supervised machine learning has proved to be significantly comprehensive in natural language processing tasks. In one particular case, we have proposed a text classification methodology, specifically used in commodity features descriptive datasets. Text including dual language characters makes it challenging to get insights into text classes. Our proposed framework employs a single language translation function to convert every item in a combination of Arabic and English strings in our dataset into English. Several preprocessing and data cleansing techniques are implemented to prepare the data to be fed into a model based on a couple of supervised machine learning algorithms. The case study introduced in this research is a typical data analytics application that many companies in production and service industry face daily. As long as descriptive data stream is loaded in databases, representing wide features of products, we are able to handle dual-lingual text, remove noise and classify each item respectively. We have addressed the problem for the parcel delivery company to categorize items and answer socio-demographics qualitative questions concerning customers and how their parcel delivery records can be monetized as valuable data source for merchandising rivals. We have gained notable precision and recall measures, proving our methodology to be responsive and accurate.

From a practical point of view, the business partner is able to monetize such a categorization function with respect to the highly desirable items specifically in online shopping. In case the vendor, the value and the frequency of sale for each commodity is available, online markets will gain more insights into sale and marketing. Other example applications would be in geo tagging parcel delivery service selection with respect to category and value of parcel.

While our solution to the commodity classification problem is based on supervised machine learning, further insight into customizations based on “Wordnet” to make it proper for similar cases and implementing the N-gram model can be further studied and the results could be interesting.

Bibliography

[Feldman, Sanger, 2006] R. Feldman. and J. Sanger. The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, 2006.

[Shankar, Lin, 2016] S. Shankar and I. Lin. Applying Machine Learning to Product Categorization. Stanford CS229, 2016.

[Kotsiantis, 2007] S.B. Kotsiantis. Supervised Machine Learning: A Review of Classification Technique. Informatica 31, 2007, 249-268 p.

[Ghani et al, 2006] Ghani et al. Text Mining for Product Attribute Extraction, SIGKDD Explorations, Volume 8, Issue 1, 41-48 p

[Popescu, Etzioni, 2005] A.M. Popescu, and O. Etzioni. Extracting Product Features and Opinions from Review. Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing (2005), 339-346 p.

[Luhn, 1957] H.P. Luhn,. A Statistical Approach to Mechanized Encoding and Searching of Literary Information. IBM Journal of research and development, IBM. 1 (4) ,1957, 315p.

[Spärck Jones, 1957] K. Spärck Jones. A Statistical Interpretation of Term Specificity and Its Application in Retrieval. Journal of Documentation 28, 1957, 11-21 p.

[Beneventano et al, 2004] Beneventano et al. A web service based framework for the semantic mapping amongst product classification schemas. Journal of Electronic Commerce Research, VOL. 5, NO.2, 2004

[Vijayarani et al, 2001] Vijayarani et al. Preprocessing Techniques for Text Mining - An Overview. International Journal of Computer Science & Communication Networks, Vol 5(1), 2001, 7-16 p.

[Aggarwal et al, 2012] Aggarwal et al. A survey of Text Classification Algorithms. Mining Text Data 2012, 163-222 p.

[Rennie et al, 2003] Rennie et al. Tackling poor assumptions of naïve Bayes classifiers. Proceedings of 20th international conference on machine learning, 2003, 616-623 p.

[Zhang, Oles, 2011] T. Zhang. and F.J. Oles Text categorization based on regularized linear classification methods. Information retrieval 4, 2011, 5-31 p.

[Joachims, 1998] T. Joachims. Text categorization with Support Vector Machines: Learning with many relevant features. Journal of Machine Learning: ECML-98,1998), 137-142

[Pang et al, 2002] Pang et al. Thumbs up?: sentiment classification using machine learning techniques. Proceedings of the ACL-02 conference on Empirical methods in natural language processing , Volume 10, 2002, 79-86 p.

[Kudo, Matsumoto, 2004] T. Kudo and Y. Matsumoto. A Boosting Algorithm for Classification of Semi-Structured Text. Conference on Empirical Methods in Natural Language Processing 2004.

[Schapire, Singer, 2000] R. Schapire and Y. Singer. BoosTexter: A Boosting-based System for Text Categorization. Machine Learning 39(2/3), 2000, 135-168 p.

[Della Pietra et al, 1997] Della Pietra et al. Inducing Features of Random Fields. IEEE Transactions Pattern Analysis and Machine Intelligence, Vol. 19, NO. 4, 1997.

[Chieu, Ng, 2002] H.L. Chieu and H.T. Ng: Named entity recognition: a maximum entropy approach using global information. Proceedings of the 19th international conference on Computational linguistics, Volume 1, 2002, 1-7 p.

[Ratnaparkhi, 1996] A. Ratnaparkhi A Maximum Entropy Model for Part-Of-Speech Tagging. Conference on Empirical Methods in Natural Language Processing 1996.

[Jurka et al, 2013] RTextTools: A Supervised Learning Package for Text Classification, The R Journal Vol. 5/1 June 2013.

Authors' Information

Aziz Yarahmadi *– PhD student at Research group of business informatics, Faculty of Business Economics, Hasselt University,B2a,Campus Diepenbeek, B-3590 Diepenbeek, Limburg, Belgium*
e-mail: aziz.yarahmadi@uhasselt.be

Major Fields of Scientific Research: Social Networks Mining, Text Mining Applications in Business, Business Process Management and Improvement

Mathijs Creemers *– PhD student at Research group of business informatics, Faculty of Business Economics, Hasselt University, B56, Campus Diepenbeek, B-3590 Diepenbeek, Limburg, Belgium*
e-mail: mathijs.creemers@uhasselt.be

Major Fields of Scientific Research: Process Mining, Knowledge Management, Resource Metrics

Hamzah Qabbaah *- PhD student at Research group of business informatics, Faculty of Business Economics, Hasselt University, B2a, Campus Diepenbeek, B-3590 Diepenbeek,Limburg,Belgium*

e-mail: hamzah.qabbaah@uhasselt.be

Major Fields of Scientific Research: Data mining, E-business, Social Network Analysis, Knowledge Management

Koen Vanhoof *- Professor Dr., Head of the discipline group of Quantitative Methods, Faculty of Business Ecnomics, Universiteit Hasselt; Campus Diepenbeek; B-3590 Diepenbeek,Limburg,Belgium*

e-mail: koen.vanhoof@uhasselt.be

Major Fields of Scientific Research: data mining, knowledge retrieval

WIRELESS SENSOR NETWORK FOR PRECISION FARMING AND ENVIRONMENTAL PROTECTION

Oleksandr Palagin, Volodymyr Romanov, Igor Galelyuka, Oleksandr Voronenko, Yuriy Brayko, Roza Imamutdinova

***Abstract**: Climate changing, anthropogenic disasters and fast spreading of viral and bacterial diseases of plants in agriculture, gardening, forestry indicates necessity to develop and full-scale production of sensor tools for express-diagnostics of plant state in real-time mode and estimating influence level of climatic changes, viral and bacterial load, and stress factors of natural and anthropogenic origin on the plants. Operating objective data about plant cover state in most cases is very important factor, which causes future strategy of keeping parklands and woodlands and proper decision making in ecological monitoring and environmental protection. For this purpose smart system of environmental monitoring based on two level wireless sensor network was proposed. Smart wireless sensors, coordinator, and concentrator as the units of the wireless sensor network, and their features, and performances are considered. Architecture of wireless sensor network, structure of applied software, and testing operation of the smart sensor network are examined.*

***Keywords**: wireless sensor network, smart sensor, coordinator, information technology.*

***ACM Classification Keywords**: H.4 Information system application*

Introduction

Unforeseen changes of climate, anthropogenic disasters and fast spreading of epidemiological viral and bacterial diseases of green plantations and agricultural plants demonstrated acute necessity to develop and full-scale production of sensor tools for express-diagnostics of plant state in real-time mode and estimating influence level of climatic changes, viral and bacterial load, stress factors of natural and anthropogenic origin on the plants.

Therefore, acquisition of live (real-time) and objective data about plant cover state in most cases is very important factor, which causes future strategy of keeping parklands and woodlands and proper decision making in ecological monitoring and environmental protection. Certainly, it would be ideal to obtain information about improvement or worsening of state of parklands or woodlands beforehand, but not after the event. It lets to avoid increasing costs on environmental protection and save plants and trees from possible loss, and also to help protect plants and trees of parklands and woodlands from viral and bacterial diseases.

In Glushkov' Institute of Cybernetics of NAS of Ukraine it was designed wireless sensor network, including wireless smart sensors, network coordinators, and concentrator. The network was tested in laboratory and field conditions and test results were examined. Hardware and software and corresponding data formats for data acquisition from wireless sensor network with unmanned aerial vehicle were developed.

Work objectives

Work objectives are development, application and testing of wireless technologies in ecological monitoring, environmental protection and precision farming.

Network of wireless smart sensors

Completion of development and manufacturing of wireless smart sensors, network coordinating nodes, network concentrator and unmanned aerial vehicle with wireless data acquisition unit made it possible to define very clearly the structure of developed network of wireless smart sensors (Fig. 1). Figure also contains the photos of real samples of wireless smart sensors, coordinating node and network concentrator.

Wireless sensors are the main network elements. The main function of wireless sensor is estimating the state of plant and further transmission of measuring data to coordinating node. Sensors measure the curve of chlorophyll fluorescence induction and store last 100 measurements in internal memory. In addition the smart sensor checks the presence of wireless connection with network coordinating node, level of battery charge and volume of free memory and, in the case of contingency, records the proper information to event log.

Network coordinating node is intended for data acquisition from smart sensors, data storing and preparation of acquired data for transmitting to user. Also, coordinating node supports the network operation, including the monitoring the functionality of smart sensors, removing the collisions during simultaneous data transmitting from several sensors, managing the measurements and informing user about state of individual sensors and network in whole.

User can acquire measuring data in one of two ways. The first way provides, that user has to apply the network concentrator, which is the protected specialized mobile computer. It is enough to approach on the distance of stable wireless connection with coordinating node of network of smart sensors. So, the coordinating node transmits all measurements to network concentrator, where the user has possibility quickly and easily to review the data and, in the case of necessity, to correct the operation of sensors or network coordinating node. In addition, the concentrator gives the possibility to run the momentary measurement of state of plants by one or all sensors.

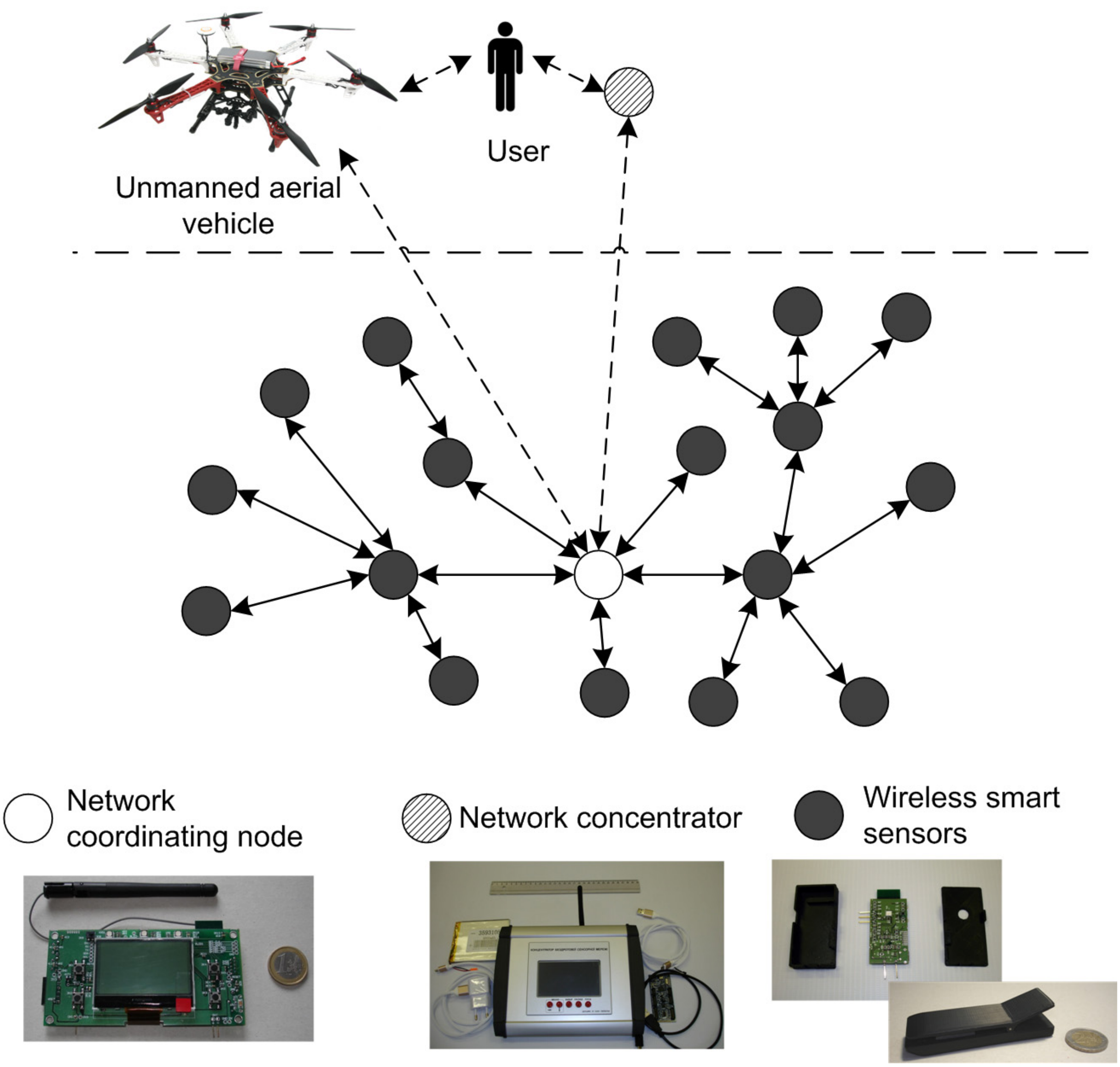


Fig. 1. Structure of network of wireless smart sensors

The second way of data acquisition may be used by user in the case, when access to network coordinating node is complicated or network coordinating node is far away from user. In this case all measuring data are acquired by unmanned aerial vehicle (hexacopter), which contains the wireless data acquisition unit with flash-memory. User inputs GPS-coordinates in flight controller of unmanned aerial vehicle, which flies to destination point and in wireless mode acquires all measuring data and event logs.

The main technical features of the smart wireless sensor are following: protection from exposure to environmental conditions at operating in real conditions; the long battery life without battery replacement or maintenance; low cost; low weight and dimensions; self-testing and self-calibration possibility for internal sensor parts; high reliability; optimum ratio between wireless data range and energy consumption; standardization and interchangeability for using electronic components; batteries replacement or charging possibility in the actual operating conditions.

The main elements that determine device functionality are wireless module and management/calculation module. The wireless microcontroller embedded into sensor significantly reduces energy consumption due to that only one microcontroller carries wireless data management, performs pre-processing of the measured data and controls input/output ports too. In addition to this approach it was applied only one software platform for the one microcontroller that in some cases significantly reduces and avoids conflicts in software. The other important hardware modules in the sensor architecture are: module which operates with biological object and power unit. The module operated with biological object would performs as the impact on biological object (irradiation leaf plants in the blue range of the spectrum) and measures the response of biological object (radiation of leaf plants in the red zone of the spectrum) on the outside influence. Power unit provides power to the all sensor's elements and enables the battery charging or replacement. Software architecture is single software application for wireless microcontrollers supported wireless data, controls and obtains measurement data from a biology object, preliminary process of measured data to storage and archiving data. The smart wireless sensor block diagram is in Fig. 2. Wireless microcontroller provides the sensor's operating, data transmission and interaction with other elements of the designed network. Since the sensor is a portable device, includes an internal battery with capacity ensured its autonomic operation for the time specified by technical requirements.

Smart wireless sensor makes it possible: to detect atmospheric air, water and soil pollutions with pesticides, heavy metals and industrial emissions; to estimate life activity of the plants after drought, frost, inoculation, pesticide application; to determine optimal dozes of chemical fertilizing and biological additions; to conduct researches in the areas of ecology and physiology of plants. Wireless smart sensor has the opportunity to operate off-line or within the network of wireless sensors. Working conditions of wireless smart sensor are following: temperature of ambient environment: from 5 till 45 ºC; relative humidity: 95% at 25 ºC; atmospheric pressure: from 84 till 107 kPa (from 630 till 800 mmHg). Technical characteristics of wireless smart sensor are following: mass of sensor: 0,03 kg; overall dimensions: 87×28×20 mm; maximum level of irradiation of leaf of living plant: 200 klux; the level of irradiation can be adjusted by coordinating node: from 50 klux to 200 klux; spectral characteristic of irradiation light: 460–480 nm; spectral characteristic of IR sensor at the level 0,5: 680–930 nm; spectral characteristic of IR sensor at the level 0,2: 550–1000 nm; irradiation area: 100 mm^2, but it is possible to increase the area according to the user requirement.

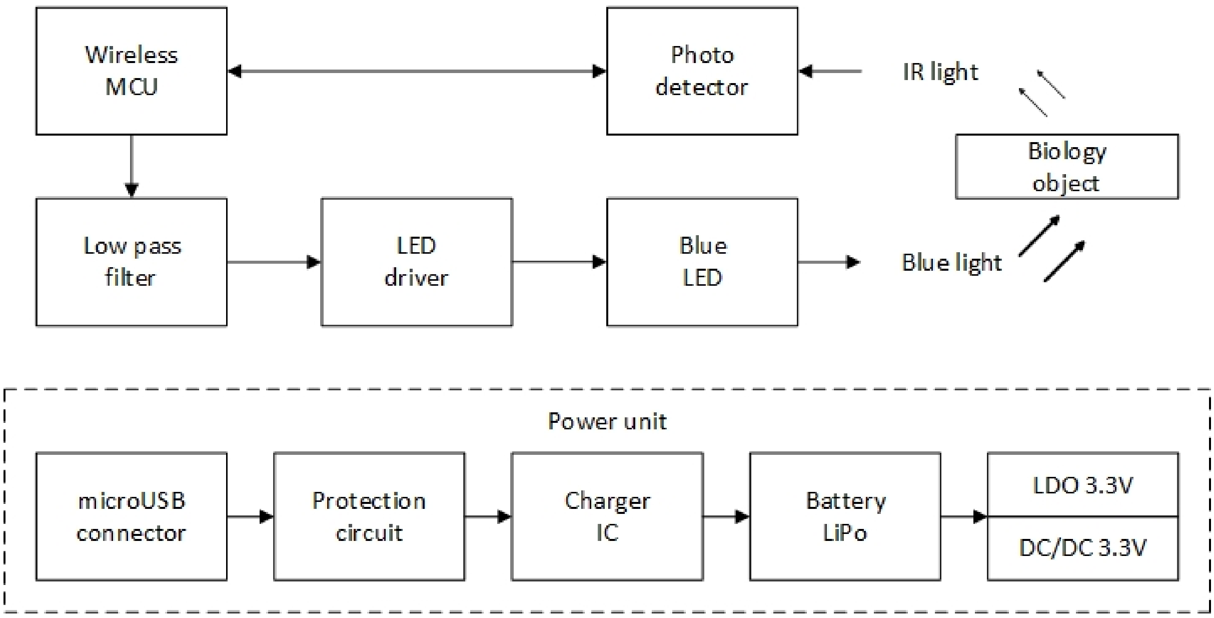
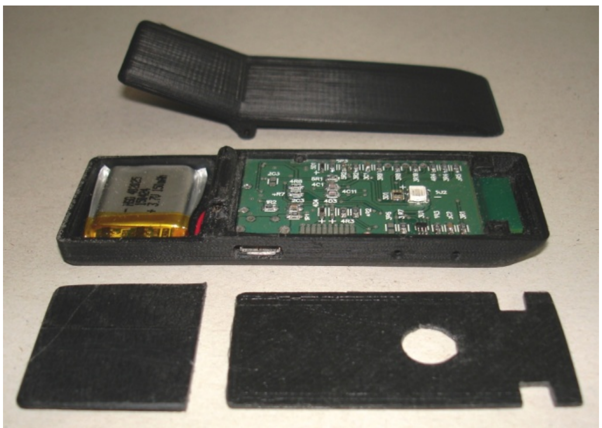
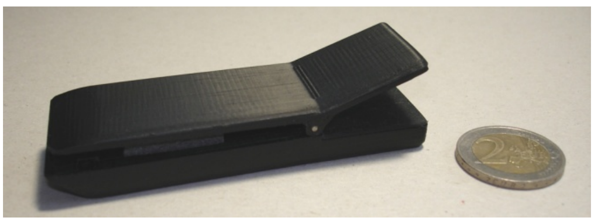


Fig. 2. Block diagram the smart wireless sensor

Assembled and disassembled wireless smart sensors are shown in Fig. 3.



a)



b)

Fig. 3. Disassembled (a) and assembled (b) wireless smart sensors

Additional functions of sensor are following: self-testing and self-calibration; control of sensor parameters in process of operating or according to the requirement of network coordinating node; measuring and control of sensor battery charging; changing emission intensity; control of emission intensity.

Performances of power supply are following: sensor includes internal lithium-pol. battery with capacity: 120–200 mAh; maximum current consumption of sensor: 50 mA; battery charging is running from external power supply from 4,3 V till 6 V through connector "microUSB"; charging current of the internal battery: 75 mA; charging time till the level 95% is approximately 2 hours; charging internal battery is running from external power supply of PowerBank, Laptop, AC/DC Adaptor through "microUSB" connector, charging voltage: 5 V–5,2 V, charging current: NLE 100 mA.

Sensor has the opportunity to receive and to transmit data if the internal battery charge is low for measuring. In process of measuring chlorophyll fluorescence induction curve the sensor measures the ambient temperature and store the temperature measuring data.

Initial time of measurement and measurement modes of chlorophyll fluorescence induction curve are initialized with coordinating node of network of sensors. The measurement modes of chlorophyll fluorescence induction curve are following: the mode of OJIP test; a preset programmed modes: 1 second, 10 seconds, 3 minutes, 4 minutes; user mode – measurement parameters are programmed by user.

It is clear, that for one user it is impossible to manually operate a large quantity of wireless sensors, which conduct simultaneous measurements in certain territory. It is necessary to introduce additional facility for joining all wireless sensors into some network for acquisition and storage of measuring data. Such facility undertakes all functions of supporting the operation of network of sensors. As additional facility it was designed and manufactured coordinating node of network for operating several of wireless sensors. The main functions of coordinating node are following: joining wireless smart sensors into network, supporting operation of the network of sensors, diagnostics of state of individual sensors and network of sensors in whole, detecting unforeseen contingencies and informing user; measuring data acquisition from all sensors and data storing; controlling the measurement process of all or selected sensors according to preset user program. The block diagram of coordinating node is shown in Fig. 4.

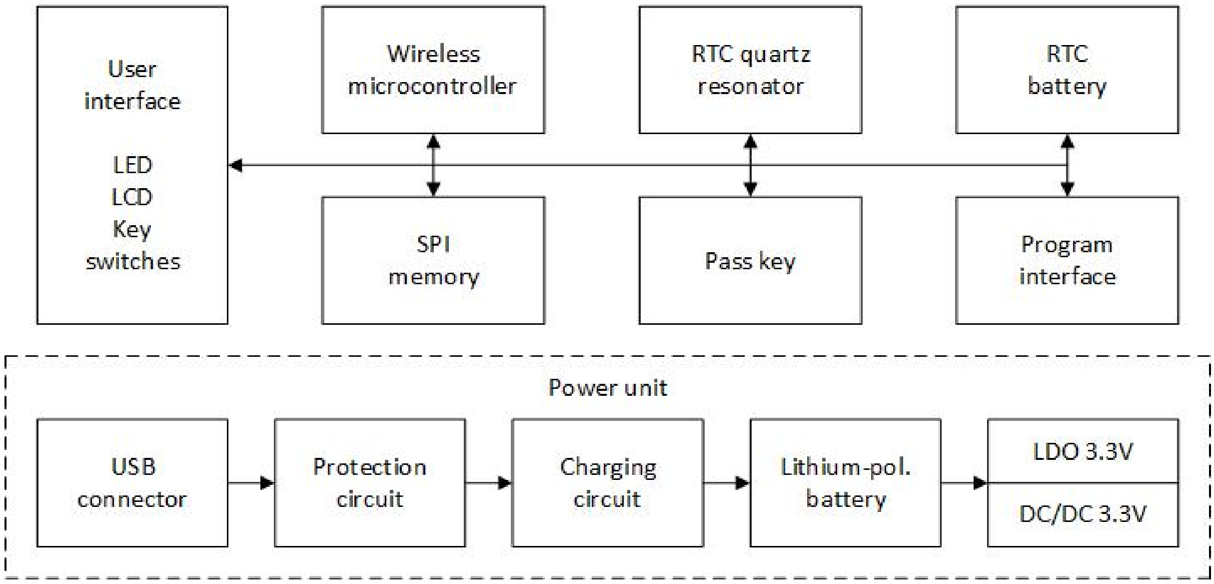


Fig. 4. Block diagram of coordinating node of network of wireless sensors

The assembled coordinator with microelectronic components is shown in Fig. 5.

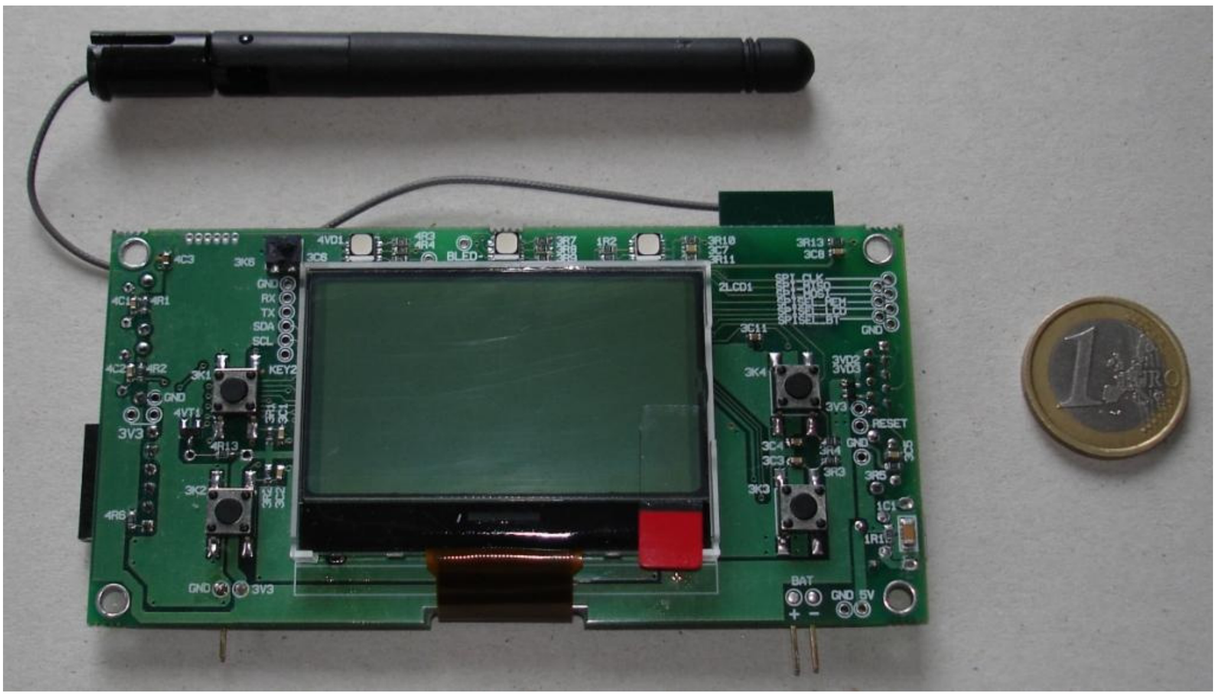


Fig. 5. Coordinating node

The block diagram of concentrator of network of wireless sensors is shown in Fig. 6. According to the diagram the concentrator includes the following units: microcontroller unit; visualisation unit (LCD); interface unit; control unit; wireless unit; debugger; DC-DC converter; charger; LiPo battery; adapter.

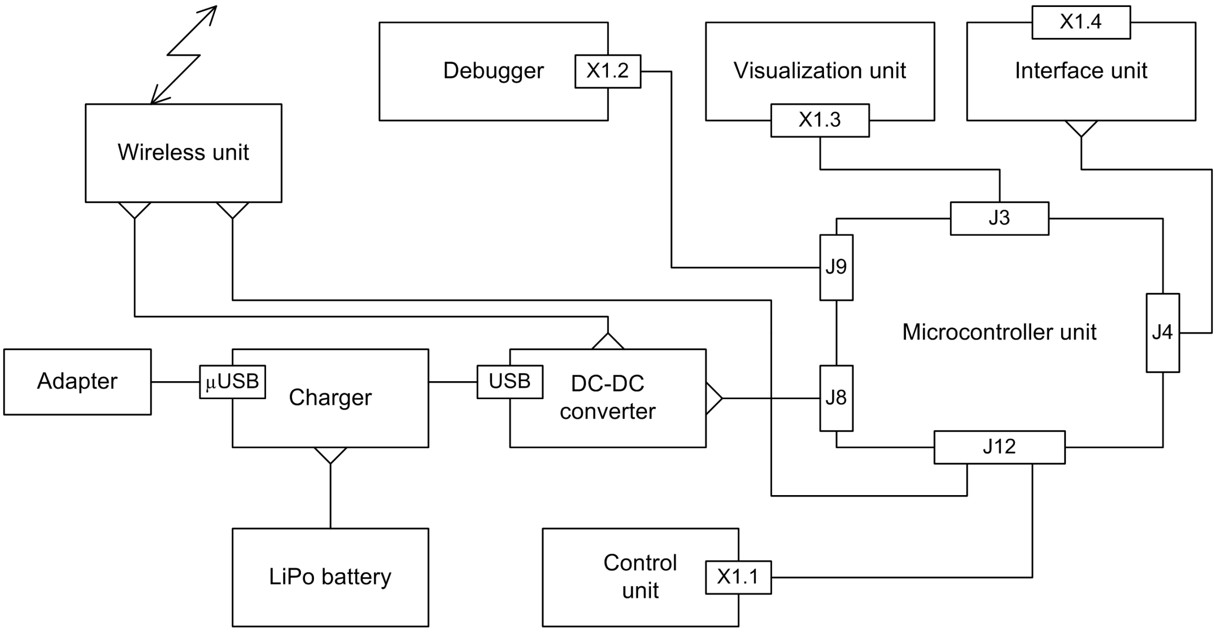


Fig. 6. The block diagram of concentrator of network of wireless sensors

The microcontroller unit is based on LPC4357. The microcontroller is intended for data acquisition, data processing, data storage and visualization of measuring data, receiving and transmitting the service information about the state of sensors and network to personal computer. The visualisation unit consists of liquid-crystal display (LCD) and serves to display the menu items and measuring data in form that is suitable for express-analysis. The interface unit is intended to connect the microcontroller unit to PC for further processing, archiving and visualization of the received information in tabular or graphical form. Control unit is designed to convert pressing of keys into signals for microcontroller. Wireless unit is based on the JN5168 microcontroller with low power consumption. This microcontroller contains a wireless module, which operates at a clock frequency of 2.4 GHz in accordance with the standard IEEE 802.15.4 (ZigBee PRO Stack). Wireless unit is used for the formation, transmission and reception of data packets from network of wireless sensors. LPC-Link2 debugger is a versatile standalone debug device, which can be connected to a microcontroller unit and supports a broad set of debugging tools and integrated development environment. Debugger is connected to concentrator, when it is necessary to reprogram and adjust the microcontroller. DC-DC converter is designed to convert the lithium-polymer battery voltage (3,7 V) in the power supply voltage of the microcontroller unit (5,0 V) with maximum current consumption of 0,5 A. The charger is used to charge the battery. Charging the LiPo battery is carried out through micro USB connector, located on the rear panel of concentrator. The charger includes protection scheme, by which the battery voltage is controlled to prevent excessive battery charging or discharging. The charger is attached to the battery and DC-DC - converter by means of cables and USB-connector. The adapter is used to convert AC 220 V to DC 5V output voltage, which is necessary for the charger operation. The concentrator of network was created as a standalone device according to modular principle. Dimensions of profile are 181,2×53,2×200 mm. The concentrator is shown in Fig. 7, and disassembled concentrator is shown in Fig. 8

Fig. 7. Concentrator of network of sensors

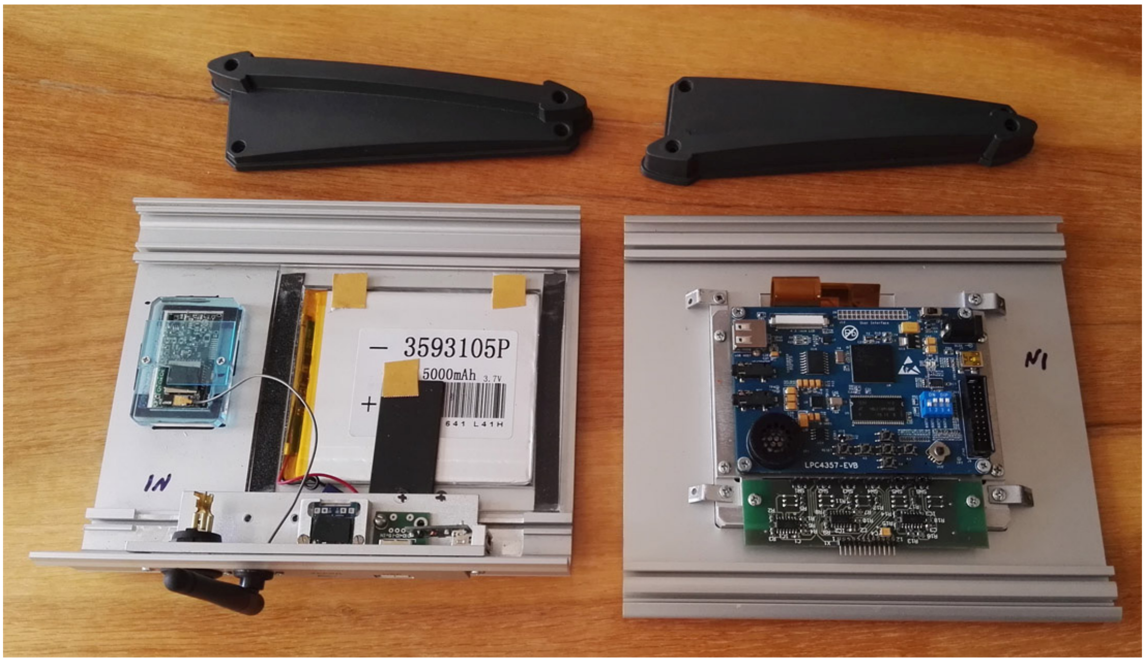


Fig. 8. Disassembled concentrator of network of sensors

LCD display FS-K430WQC-V3-F, based on graphics controller SSD1963, is included in the visualization block to display measuring data and service information. For user convenience the display has diagonal of 11 cm with 480 x 272 pixels resolution. The number of displayed colors is 16 700. The display has low power consumption, and typical current consumption is equal to 15 mA. On front panel of concentrator there are display of visualization unit and five buttons of control unit: MENU (right), MENU (down), SELECT, BACK, START, which are intended to navigate menu.

Proper applied software was developed for every hardware component of network in accordance with developed structure. Structure of developed software and interaction of program modules are shown on Fig. 9.

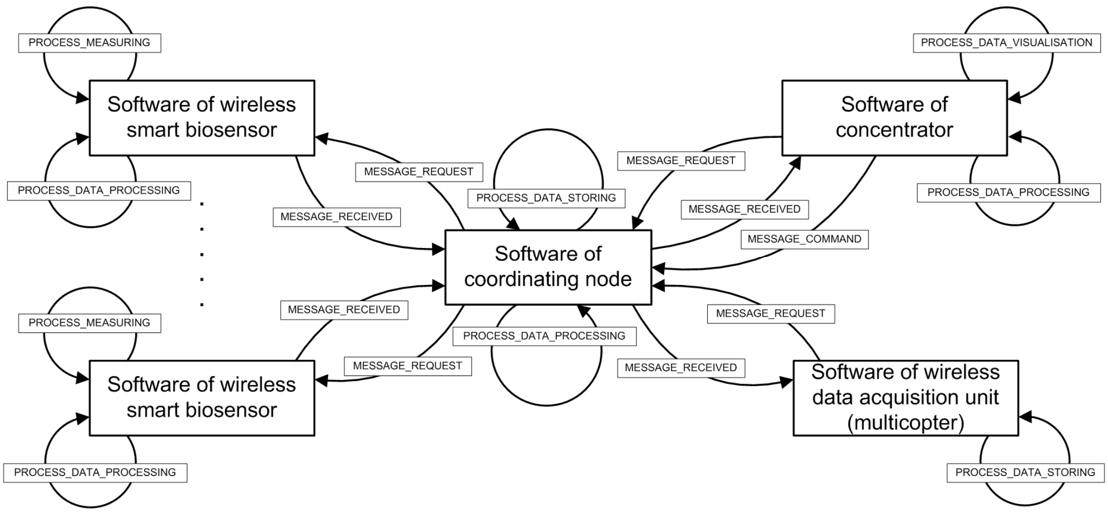


Fig. 9. Structure of applied software of network of wireless smart sensors

Applied software of wireless smart sensor ensures continuous running of sensor in autonomous power-saving mode in field conditions and provides the next functions: searching the network coordinating node and connecting to network; measuring the curve of chlorophyll fluorescence induction of plant and storing measuring data; storing the preset number of measurements; recurring diagnostics of its state, level of battery charge, volume of free memory, detection of unforeseen events and logging; preparing the measuring data and event log for transmission; transmission of data on request of coordinating node; conducting the preset measurement program or instantaneous measurement on request of user.

After manufacture of the units of wireless sensor networks they were tested in laboratory and field conditions. Main attention was paid to reliability of sensor operation in field conditions and stability of wireless data transfer channel in different conditions of environment. Data, obtained during analysis of test results, were compared with modeling results. At the first, it was tested the reliability of message transferring or, in another words, the quality of wireless channel between wireless sensors and network coordinating node. Tests were conducted for different conditions of environment, which were interesting for prospective customers. There were 4 types of environment during testing: open territory, where there was line of sight between signal source and signal receiver; low vegetation, grass, many bushes; dense vegetation, many high trees, line of sight between signal source and signal receiver was absent in most cases; urban territory, line of sight between signal source and signal receiver was absent. The results of testing are presented in Fig. 10.

For second type of testing of parameters of wireless smart sensors in laboratory conditions it was used applied software "JN516x Customer Module Evaluation Tool", which is delivered by NXP Company for testing its wireless microcontrollers, and digital-to-analogue analyzer "LabTool" with proper applied software. This testing was conducted in laboratory conditions for estimating real time, which wireless sensor spends for data transmission. This time, to some extent, defines the time of autonomous operation of wireless sensor without battery replacement. Internal timer of JN5168 microcontroller determined the time of wake-up of module and time of receiving of acknowledgment of successful transmission of one data frame. There were transmitted data frames, payloads of which was divisible 114 bytes. Ten measurements of time were made for every data package size and then average time was calculated for these 10 values. Dependence of experimental average time of transmission of data packages on data package size are shown together with calculated values in Fig. 11.

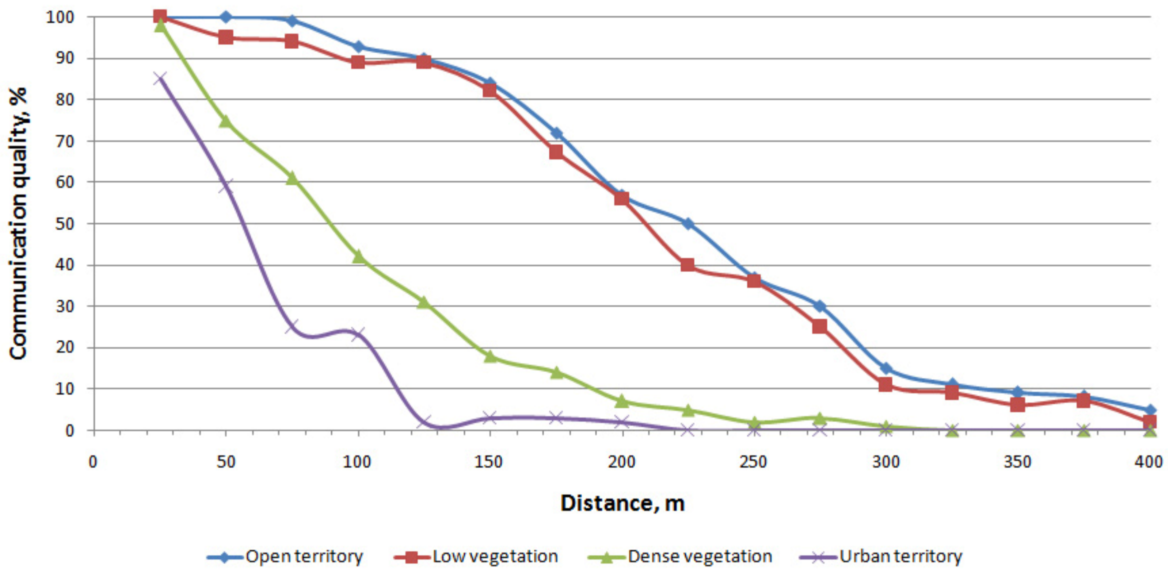


Fig. 10. Dependence of communication quality on distance between signal source and signal receiver

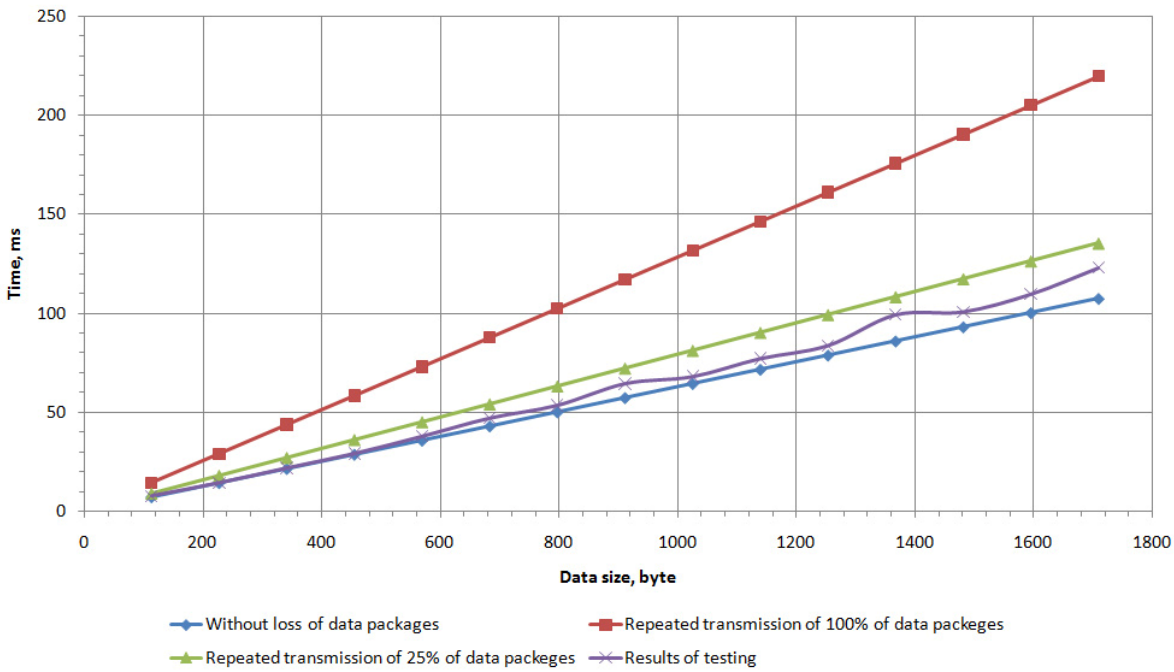


Fig. 11. Dependence of transmission time of data package on data package size

The next testing was conducted to estimate the operation time of wireless sensors in autonomous mode without battery replacement. Sensor is in sleep mode most of time. One time per three seconds the sensor wakes from sleep and checks the accessibility of wireless network and presence of commands from coordinating node. After waking-up the sensor initializes its hardware and program modules. If network is accessible and command buffer is empty, the sensor again falls asleep. If command buffer contains commands from coordinating node, the sensor measures the plant state or diagnoses the wireless channel state, level of battery charge and volume of free memory. If sensor operates according to preset user program, the plant state is measured in defined time and measuring data are transmitted to coordinating node. Measurement of power-consumption of sensor on every stage of its operation was conducted by means of software tool "JN516x Customer Module Evaluation Tool" and digital-to-analogue analyzer "LabTool" with proper applied software. Wireless smart sensor operates from the battery with capacity of 200 mAh. Packert exponent or, in another words, coefficient of incompleteness of discharge of battery is equal to 1,3. First of all, the measurement of power-consumption of sensor was conducted for next operational modes "Wake from sleep – initialization – channel selection – clear channel assessment – sleep". We obtained the next results: wake from sleep: supply current – 4,98 mA, period – 0,83 ms; initialization: supply current – 5 mA, period – 1 ms;c channel selection: supply current – 5,16 mA, period – 0,96 ms; clear channel assessment: supply current – 20,28 mA, period – 0,128 ms; sleep: supply current – 0,00064 mA, period – 180 000 ms.

On the whole, the average supply current during all modes was equal to 6,21 μA. In such mode without conducting any measurements the wireless sensor can operate for more than 2,5 years without battery replacement.

In measuring mode two new modes is added to mentioned above operational modes: measuring and data transmission. It should be noted, that measurement can last 1 second, 10 seconds, 3 or 4 minutes on user choice. Also, it can be conducted one or several measurement per day, or one measurement per several days. Testing showed, that supply current, needed for measurement, was equal to 25 mA. For example, for one measurement during 10 seconds per day the average supply current during 24 hours was equal to 9,1 μA, and for one measurement during 3 minutes per day – 58 μA. Then by using the calculation it was estimated the time of autonomous operation of sensor for certain number of measurements per day. Experimental testing and calculations let to obtain the dependence of time of autonomous operation of sensor without battery replacement on number of measurements per day. The results of testing are shown in Fig. 12.

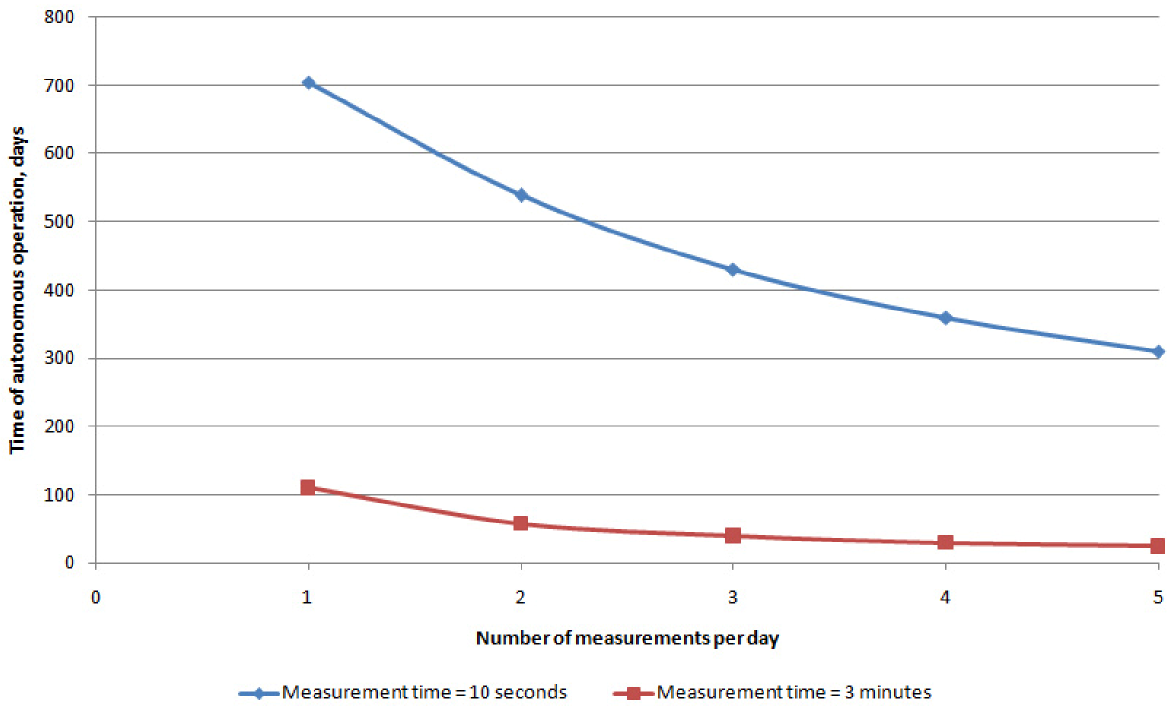


Fig. 12. Dependence of time of autonomous operation of sensor without battery replacement on number of measurements per day

Development and assembly of unmanned aerial vehicle (in our case this is hexacopter) with wireless data acquisition unit consists of 3 stages: assembly and adjustment of hexacopter, including adjustment and programming the flight controller together with GPS-unit and compass; development and manufacture of hardware of wireless data acquisition unit; development of software of wireless data acquisition unit.

The hexacopter on the basis of chassis "DJI F550" and engines "DJI 2212" was chosen for the second level of network. Controller "DJI NAZA-M V2" with GPS-unit and compass was chosen as flight controller. Ready-assembled hexacopter is shown on Fig. 13.

For using hexacopter in the area of ecological monitoring and environmental protection it is necessary to ensure the flight to preset GPS-coordinates in automatic mode and then returning home. In another words, the hexacopter has to fly to point with GPS-coordinates, where the network coordinating node is situated, acquire measuring data and return home. Setting flight route by means of GPS-coordinates is fulfilled in any special third-party software.

Fig. 13. Ready-assembled unmanned aerial vehicle

The wireless unit of developed sensor with limited functionality was used as basis of hardware of wireless data acquisition unit for hexacopter. There are main functions of wireless data acquisition unit: establishing connection with pre-given network during flight at distance, enough for stable wireless connection; sending request to coordinating node for data transmission; receiving all data from coordinating node and storing them in internal memory. Appearance of developed wireless data acquisition unit is shown on Fig. 14. On the left side of photo one can see USB-connector to transfer data to personal computer, on the right side – antenna for establishing wireless communication.

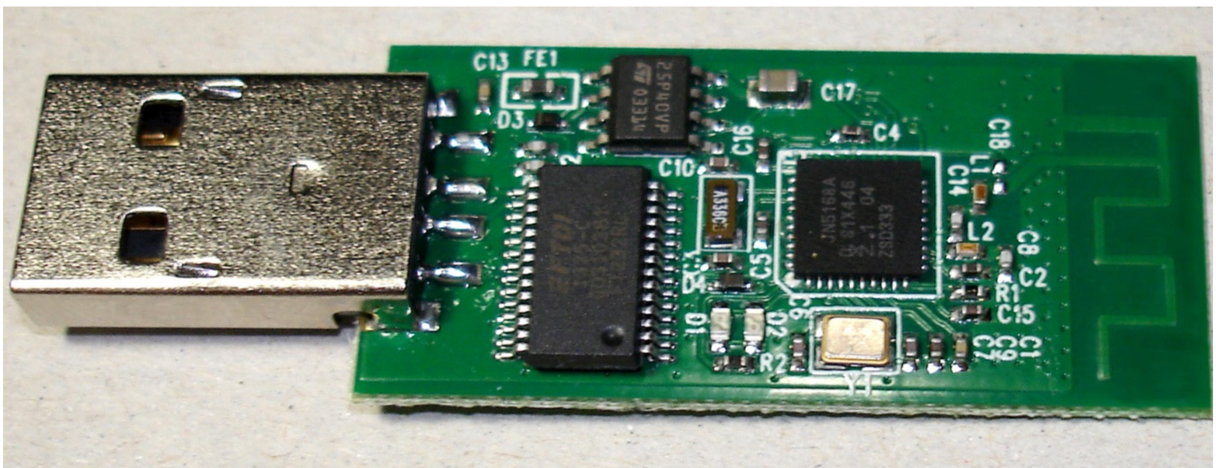


Fig. 14. Wireless data acquisition unit

Conclusion

During STCU project # 6064 "Developing and full-scale production preparing of distributed smart biosensors for environmental protection" it was designed and manufactured two-level network of wireless sensors included wireless smart sensors, coordinating and concentrating nodes with applied and testing software and mobile platform based on unmanned aerial vehicle such as hexacopter.

Acknowledgement

The paper is published with partial support by the project ITHEA XXI of the ITHEA ISS (www.ithea.org) and the ADUIS (www.aduis.com.ua).

Authors' Information

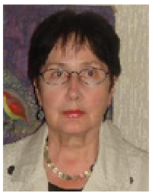
Oleksandr Palagin – *Depute-director of V.M.Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine, Academician of National Academy of Sciences of Ukraine, Doctor of technical sciences, professor; Prospect Akademika Glushkova 40, Kiev, 03187, Ukraine; e-mail: palagin_a@ukr.net*

Volodymyr Romanov – *Head of department of V.M.Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine, Doctor of engineering sciences, professor; Prospect Akademika Glushkova 40, Kiev, 03187, Ukraine; e-mail: VRomanov@i.ua; website: http://www.dasd.com.ua*

Igor Galelyuka – *Senior research fellow of V.M.Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine; Candidate of engineering sciences; Prospect Akademika Glushkova 40, Kiev, 03187, Ukraine; e-mail: galelyuka@gmail.com; website: http://www.dasd.com.ua*

Oleksandr Voronenko – *research fellow of V.M.Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine; Prospect Akademika Glushkova 40, Kiev, 03187, Ukraine; e-mail: alexander.voronenko@dci.kiev.ua; website: http://www.dasd.com.ua*

Yuriy Brayko – *research fellow of V.M. Glushkov’s Institute of Cybernetics of National Academy of Sciences of Ukraine; Prospect Akademika Glushkova 40, Kiev, 03187, Ukraine; e-mail: yrbray@mail.ru; website: http://www.dasd.com.ua*



Roza Imamutdinova – *programmer of V.M. Glushkov’s Institute of Cybernetics of National Academy of Sciences of Ukraine; Prospect Akademika Glushkova 40, Kiev, 03187, Ukraine; e-mail: rozima@mail.ru; website: http://www.dasd.com.ua*

DEEP HYBRID SYSTEM OF COMPUTATIONAL INTELLIGENCE FOR TIME SERIES PREDICTION

Yevgeniy Bodyanskiy, Iryna Pliss, Dmytro Peleshko, Olena Vynokurova

Abstract: *In this paper the deep hybrid system of computational intelligence for time series forecasting is proposed. As the layer of such hybrid system we propose to use the hybrid generalized additive type-2 fuzzy-wavelet-neural network. Proposed deep stacking forecasting hybrid system of computational intelligence is enough simple in computational implementation due to parallelizing process of implemented computations, has high learning rate convergency due to dismissal from errors backpropagation and using the rapid adaptive algorithms for tuning its parameters, has also flexibility due to possibility of tuning the activation-membership functions. Proposed system is aimed for solving the wide range Data Stream Mining tasks, such as forecasting, identification, emulation, classification and pattern recognition in online mode.*

Keywords: *deep learning, computational intelligence, time series forecasting, hybrid type-2 fuzzy system.*

ACM Classification Keywords: *I.2 ARTIFICIAL INTELLIGENCE, I.5 PATTERN RECOGNITION.*

Introduction

Nowadays hybrid systems of computational intelligence [Rutkowski, 2008; Du and Swamy, 2014; Mumfold and Jain, 2009] are wide spread for solving different type of Data Mining problems under uncertainty conditions when the investigated objects or processes are nonstationary, nonlinear, stochastic, chaotic, etc. Among these problems the most complicated task is forecasting because it has highest level of uncertainty, which is defined by multivariance of future facts and events. Specifically, hybrid systems have shown the best results for solving such type of tasks due to universal approximation properties and capability to learning of artificial neural networks [Haykin, 1999; Nelles, 2001], transparency and interpretability of obtained results and high learning rate of tuning synaptic weights of neuro-fuzzy systems [Jang and Mizutani, 1997; Bodyanskiy et all, 2001; Bodyanskiy et all, 2005; Bodyanskiy et all, 2008], including type-2 fuzzy neural networks [Aliev and Guirimov, 2014], the possibility of effective description of nonstationary signal local characteristics due to wavelet neural networks [Alexandridis and Zapranis, 2014] and flexibility of wavelet-neuro-fuzzy systems [Abiyev and

Kaynak, 2008; Bodyanskiy and Vynokurova, 2013], including type-2 fuzzy-wavelet neural networks [Bodyanskiy et all, 2015].

Recent years deep neural networks [LeCun et all, 2015; Schmidhuber, 2015; Goodfellow et all, 2016] attract the attention of many researchers. There networks provide higher quality of data processing including forecasting problem [Langkvist Längkvist et all, 2014; Huang et all, 2014; Setlak et all, 2016] in comparison with the conventional shallow neural networks. But the learning process of deep networks is enough time-consuming and demands heavy span time, leading to impossibility of information processing in real time. Nevertheless, such tasks are appeared quite often, especially in the case when data are fed to the processing in on-line mode and time for processing is strictly constrained. These problems have appeared in the context of Dynamic Data Mining [Lughofer, 2011] and Data Stream Mining [Aggarwal, 2007; Bifet, 2010] and even accelerated deep systems [Bodyanskiy et all, 2016], which are constructed based on conventional architectures, cannot manage to adjust their parameters in a time between two next observations.

In the connection with that topical problem is synthesis of forecasting deep hybrid system of computational intelligence, which allows to process the nonlinear nonstationary time series in on-line mode.

Architecture of deep forecasting hybrid system

Figure 1 shows the architecture of proposed forecasting deep systems of computational intelligence, which is a hybrid of cascade-correlation neural network of Fahlman-Lebiere [Fahlman and Lebiere, 1990], cascaded neuro-fuzzy systems [Bodyanskiy et all, 2014; Bodyanskiy and Tyshchenko, 2016; Bodyanskiy et all, 2016a], stacking deep neural networks [Schmidhuber, 2015] and hybrid generalized additive type-2 fuzzy-wavelet-neural network [Bodyanskiy et all, 2015], that form layers of deep system. At that number of layers can be increased during learning process of system.

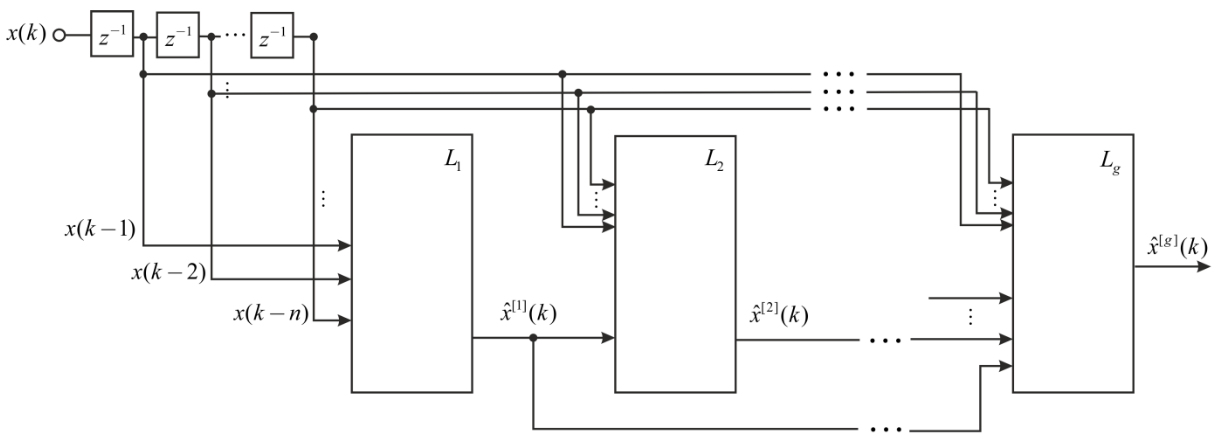


Figure 1. Forecasting deep hybrid system of computational intelligence

The zero layer of system is formed by elements with pure delay z^{-1}, at that the current observation of processed time series $x(k)$ is fed to input of first element. Thus, the zero layer forms prehistory of process $X(k)=(x(k-1),x(k-2),\ldots,x(k-n))^T \in R^n$, which is fed to input of the first hidden layer L_1. The output of this layer is estimated $\hat{x}^{[1]}(k)=f^{[1]}(X(k))$, where $f^{[1]}(X(k))$ is nonlinear transformation which is implemented by HGAT2FWNN [Bodyanskiy et all, 2015]. Input of second layer L_2 is signal $X^{[2]}(k)=(X^T(k),\hat{x}^{[1]}(k))^T \in R^{n+1}$, and its output is $\hat{x}^{[2]}(k)$; to the input of third layer L_3 the signal $X^{[3]}(k)=(X^T(k),\hat{x}^{[1]}(k),\hat{x}^{[2]}(k))^T=(X^{[2]T}(k),\hat{x}^{[2]}(k))^T \in R^{n+2}$ is fed, and finally, the vector $X^{[g]}(k)=(X^T(k),\hat{x}^{[1]}(k),\hat{x}^{[2]}(k),\ldots,\hat{x}^{[g-1]}(k))^T=(X^{[g-1]T}(k),\hat{x}^{[g-1]}(k))^T \in R^{n+g-1}$ is fed to input of $g-$th layer L_g, and the signal-forecast $\hat{x}^{[g]}(k)$ is formed in the output of system. Such architecture allows to avoid using the errors backpropagation procedures, which are implemented in batch mode and doesn't allow to realize online data processing. In our case the learning process is involved sequentially layer by layer and each layer is learned by high-speed adaptive algorithms. At that each next layer is distinct from previous layer only additional input, and output signal from previous layer is fed to this input.

Hybrid generalized additive type-2 fuzzy-wavelet-neural network

Figure 2 shows the architecture of hybrid generalized additive type-2 fuzzy-wavelet-neural network (HGAT2FWNN) in first hidden layer L_1 with n inputs $x_1(k)\equiv x(k-1),\ x_2(k)\equiv x(k-2),\ldots,x_n(k)\equiv x(k-n)$ and single output $\hat{x}^{[1]}(k)$.

This system consists of four layers of information processing; the first and second layers are similar to the layers of TSK-neuro-fuzzy system [Takagi and Sugeno, 1985; Sugeno and Kang, 1988; Takagi and Hayashi, 1991]. The only difference is that the odd wavelet membership functions "Mexican Hat", which are "close relative" of Gaussians, are used instead of conventional bell-shaped Gaussian membership function in the first hidden layer

$$\phi_{li}(x_i(k))=(1-\tau_{li}^2(k))\exp(-\tau_{li}^2(k)/2) \qquad (1)$$

where $\tau_{li}(k)=(x_i(k)-c_{li})\sigma_{li}^{-1}$; c_{li},σ_{li} are the centre and width of the corresponding membership function implying that $\underline{c}\le c_{li}\le\overline{c}$; $\underline{\sigma}\le\sigma_{li}\le\overline{\sigma}$; $i=1,2,\ldots,n$; $l=1,2,\ldots,h$; n is the input number; h is the membership function number.

It is necessary to note that using the wavelet functions instead of common bell-shaped positive membership functions gives the system more flexibility [Mitaim and Kosko, 1996], and using odd wavelets for the fuzzy reasoning does not contradict the ideas of fuzzy inference, because the negative values of these functions can be interpreted as non-membership levels [Mitaim and Kosko, 1997].

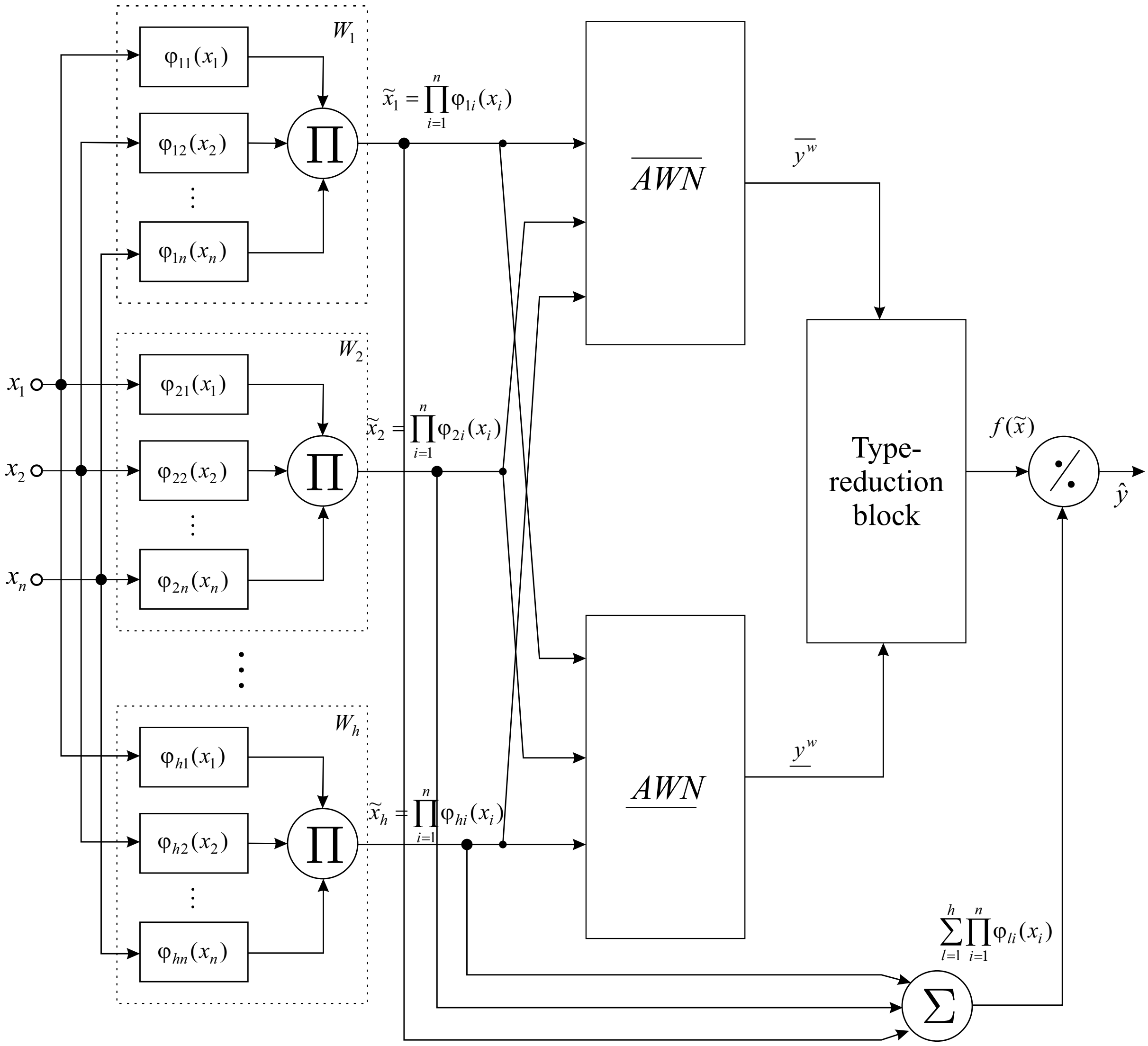


Figure 2. Architecture of HGAT2FWNN

Thus, if the input vector $X(k)$ is fed to the system input, then in the first layer the hn levels of membership functions $\phi_{li}(x_i(k))$ are computed and in the hidden layer h vector product blocks perform the aggregation of these memberships in the form

$$\tilde{x}_l(k) = \prod_{i=1}^{n} \phi_{li}(x_i(k)) . \tag{2}$$

This means that the input layers of HGAT2FWNN transform the information similarly to the neurons of the wavelet neural networks [Alexandridis and Zapranis, 2014; Bodyanskiy et all, 2005a], which form the multidimensional activation functions

$$\prod_{i=1}^{n}(1-\tau_{li}^{2}(k))\exp(-\tau_{li}^{2}(k)/2) \quad (3)$$

providing a scatter partitioning of the input space.

As a result, the signals in the output of the second layer can be written in the form

$$\tilde{x}_l(k)=\prod_{i=1}^{n}\left(1-\frac{(x_i(k)-c_{li})^2}{\sigma_{li}^2}\right)\exp\left(-\frac{(x_i(k)-c_{li})^2}{2\sigma_{li}^2}\right). \quad (4)$$

To provide the required approximation properties, the third layer of system is formed based on type-2 fuzzy wavelet neuron (T2FWN) [Bodyanskiy and Vynokurova, 2011; Bodyanskiy et all, 2012]. This neuron consists of two adaptive wavelet neurons (AWN) [Bodyanskiy et all, 2005a; Bodyanskiy et all, 2014a], whose prototype is a wavelet neuron of T. Yamakawa [Yamakawa et all, 1998]. Wavelet neuron differs from the popular neo-fuzzy neuron [Yamakawa et all, 1992; Uchino and Yamakawa, 1997] that uses the odd wavelet functions instead of the common triangular membership functions. The use of odd wavelet membership functions, which form the wavelet synapses $WS_1,\ldots,WS_l,\ldots,WS_h$, provides higher quality of approximation in comparison with nonlinear synapses of neo-fuzzy neurons. Figure 3 shows the architecture of wavelet-neuron.

In such a way the wavelet neuron performs the mapping in the form

$$y^w(\tilde{x}(k))=\sum_{l=1}^{h}f_l(\tilde{x}_l(k)) \quad (5)$$

where $\tilde{x}(k)=(\tilde{x}_1(k),\ldots,\tilde{x}_l(k),\ldots,\tilde{x}_h(k))^T$, $y^w(\tilde{x}(k))$ is the scalar output of wavelet neuron. Each wavelet synapse WS_l consists of p wavelet membership functions $\tilde{\phi}_{jl}(\tilde{x}_l)$, $j=1,2,\ldots,p$ (p is a wavelet membership function number in the wavelet neuron) and the same number of the tuning synaptic weights w_{jl} . Thus, the transform that is implemented by each wavelet synapse WS_l in the k-th instant of time, can be written in form

$$f_l(\tilde{x}_l(k))=\sum_{j=1}^{p}w_{jl}(k-1)\tilde{\phi}_{jl}(\tilde{x}_l(k)) \quad (6)$$

(here $w_{jl}(k-1)$ is the value of synaptic weights that are computed based on previous $k-1$ observations), and the general wavelet neuron performs the nonlinear mapping in the form

$$y^w(\tilde{x}(k))=\sum_{l=1}^{h}\sum_{j=1}^{p}w_{jl}(k-1)\tilde{\phi}_{jl}(\tilde{x}_l(k)) \quad (7)$$

i.e., in fact, this is the generalised additive model [Hastie and Tibshirani, 1990] that is characterised by the simplicity of computations and high approximation properties.

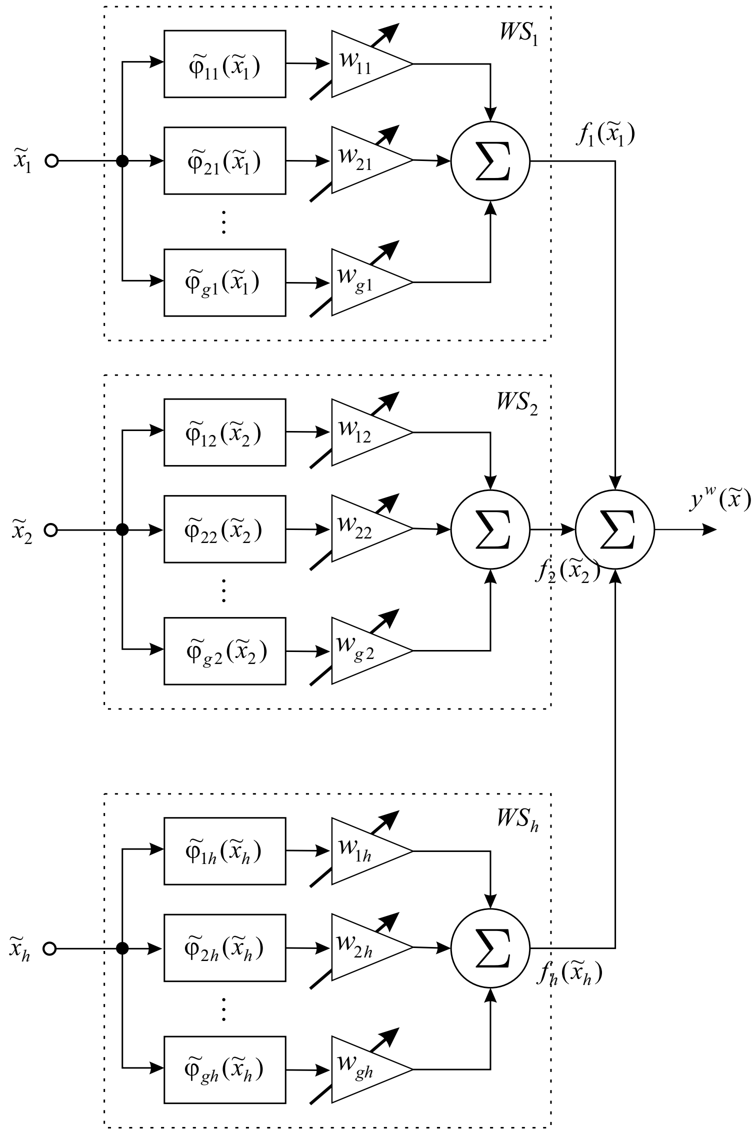
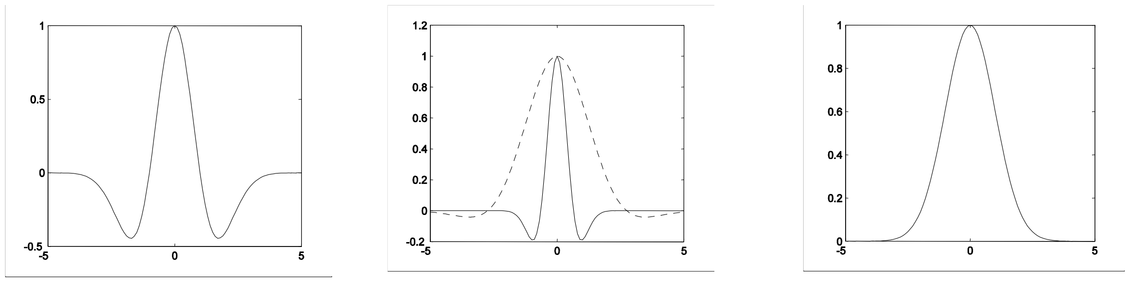


Figure 3. Wavelet-neuron architecture

Under uncertain, stochastic or chaotic conditions, it is more effective to use the adaptive wavelet neuron (AWN) instead of common wavelet neuron. The adaptive wavelet neuron is based on the adaptive wavelet function in the form

$$\tilde{\phi}_{jl}(\tilde{x}_l(k)) = (1-\alpha_{jl}\tau_{jl}^2(k))\exp(-\tau_{jl}^2(k)/2) = \left(1-\alpha_{jl}\frac{(\tilde{x}_l(k)-c_{jl})^2}{\sigma_{jl}^2}\right)\exp\left(-\frac{(\tilde{x}_l(k)-c_{jl})^2}{2\sigma_{jl}^2}\right) \quad (8)$$

where $0 \le \alpha_{jl} \le 1$ is the shape parameter of adaptive wavelet function, if $\alpha_{jl} = 0$ it is conventional Gaussian, if $\alpha_{jl} = 1$ it is the wavelet "Mexican Hat", and if $0 < \alpha_{jl} < 1$ it is some hybrid activation-membership function (see Figure 4).

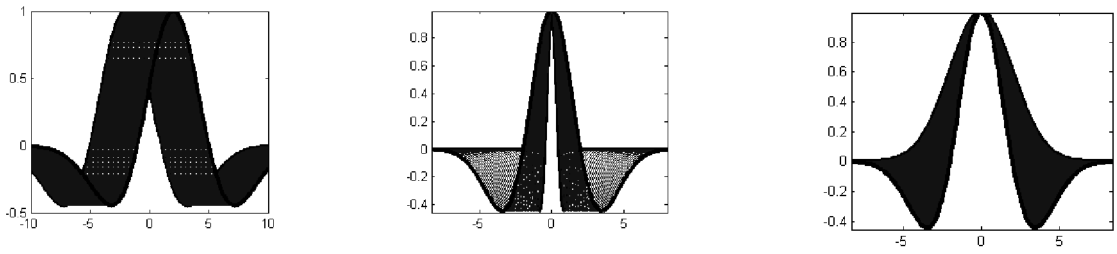


a) b) c)

Figure 4. Adaptive wavelet function

Here it should be noted that if the learning process of wavelet neuron of T. Yamakawa is the tuning of the synaptic weights w_{jl}, then the learning process of adaptive wavelet neuron consists of the tuning not only synaptic weights but also centres c_{jl}, widths σ_{jl} and shape parameters α_{jl} of wavelet functions. However, if synaptic weights w_{jl} can be tuned using the second-order optimisation algorithms, such as a recurrent least squares method, then the optimisation of the operation speed in the gradient learning algorithms of c_{jl}, σ_{jl}, α_{jl} is significantly difficult.

To overcome this difficulty, we can set the boundary of possible changes of adaptive wavelet function parameters $\underline{c}_{jl} \le c_{jl} \le \bar{c}_{jl}$, $\underline{\sigma}_{jl} \le \sigma_{jl} \le \bar{\sigma}_{jl}$, $\underline{\alpha}_{jl} \le \alpha_{jl} \le \bar{\alpha}_{jl}$ and introduce the type-2 fuzzy wavelet membership functions. These functions form the type-2 fuzzy wavelet neuron (T2FWN) and are shown in Figure 5.



a) uncertain centre b) uncertain width c) uncertain function shape parameter

Figure 5. Type-2 fuzzy-wavelet membership function with different type uncertainties

T2FWN consists of two AWNs, where the signal $\tilde{x}(k)$ is fed to the inputs of it. Thus, one AWN uses low boundary values $\underline{c}_{jl}$, $\underline{\sigma}_{jl}$, $\underline{\alpha}_{jl}$, while the other AWN uses high boundary values $\overline{c}_{jl}$, $\overline{\sigma}_{jl}$, $\overline{\alpha}_{jl}$. It is important to notice that neurons $\underline{AWN}$, $\overline{AWN}$ are trained independently of each other based on common reference signal $x(k)$.

As a result, the transformation that is implemented by $\underline{AWN}$ can be written in the form

$$\underline{y}^w(\tilde{x}(k)) = \sum_{l=1}^{h}\sum_{j=1}^{p}\underline{w}_{jl}(k-1)\underline{\tilde{\phi}}_{jl}(\tilde{x}_l(k)) \tag{9}$$

and $\overline{AWN}$ -

$$\overline{y}^w(\tilde{x}(k)) = \sum_{l=1}^{h}\sum_{j=1}^{p}\overline{w}_{jl}(k-1)\overline{\tilde{\phi}}_{jl}(\tilde{x}_l(k)) \tag{10}$$

where

$$\underline{\tilde{\phi}}_{jl}(\tilde{x}_l(k)) = \left(1-\underline{\alpha}_{jl}\frac{(\tilde{x}_l(k)-\underline{c}_{jl})^2}{\underline{\sigma}_{jl}^2}\right)\exp\left(-\frac{(\tilde{x}_l(k)-\underline{c}_{jl})^2}{2\underline{\sigma}_{jl}^2}\right), \tag{11}$$

$$\overline{\tilde{\phi}}_{jl}(\tilde{x}_l(k)) = \left(1-\overline{\alpha}_{jl}\frac{(\tilde{x}_l(k)-\overline{c}_{jl})^2}{\overline{\sigma}_{jl}^2}\right)\exp\left(-\frac{(\tilde{x}_l(k)-\overline{c}_{jl})^2}{2\overline{\sigma}_{jl}^2}\right). \tag{12}$$

In the type-reduction block, the signals $\underline{y}^w(\tilde{x}(k))$ and $\overline{y}^w(\tilde{x}(k))$ are united in the simplest way and form the output signal of T2FWN

$$f(\tilde{x}(k)) = c(k)\overline{y}^w(\tilde{x}(k)) + (1-c(k))\underline{y}^w(\tilde{x}(k)) \tag{13}$$

where $c(k)$ is the tuning parameter that defines closeness of signals $\underline{y}^w(\tilde{x}(k))$ and $\overline{y}^w(\tilde{x}(k))$ to reference signal $x(k)$.

Finally, the forth (output) layer of system that consists of elementary sum and division blocks implements the defuzzification in the form

$$\hat{x}^{[1]}(k) = \frac{f(\tilde{x}(k))}{\sum_{l=1}^{h}\tilde{x}_l(k)} = c(k)\frac{\overline{y}^w(\tilde{x}(k))}{\sum_{l=1}^{h}\tilde{x}_l(k)} + (1-c(k))\frac{\underline{y}^w(\tilde{x}(k))}{\sum_{l=1}^{h}\tilde{x}_l(k)} = \tag{14}$$

$$= c(k)\frac{\sum_{l=1}^{h}\sum_{j=1}^{p}\bar{w}_{jl}(k-1)\bar{\tilde{\phi}}_{jl}(\tilde{x}_l(k))}{\sum_{l=1}^{h}\tilde{x}_l(k)} + (1-c(k))\frac{\sum_{l=1}^{h}\sum_{j=1}^{p}\underline{w}_{jl}(k-1)\underline{\tilde{\phi}}_{jl}(\tilde{x}_l(k))}{\sum_{l=1}^{h}\tilde{x}_l(k)} =$$

$$= c(k)\sum_{l=1}^{h}\sum_{j=1}^{p}\bar{w}_{jl}(k-1)\frac{\bar{\tilde{\phi}}_{jl}(\tilde{x}_l(k))}{\sum_{l=1}^{h}\tilde{x}_l(k)} + (1-c(k))\sum_{l=1}^{h}\sum_{j=1}^{p}\underline{w}_{jl}(k-1)\frac{\underline{\tilde{\phi}}_{jl}(\tilde{x}_l(k))}{\sum_{l=1}^{h}\tilde{x}_l(k)} =$$

$$= c(k)\sum_{l=1}^{h}\sum_{j=1}^{p}\bar{w}_{jl}(k-1)\bar{\tilde{\psi}}_{jl}(\tilde{x}_l(k)) + (1-c(k))\sum_{l=1}^{h}\sum_{j=1}^{p}\underline{w}_{jl}(k-1)\underline{\tilde{\psi}}_{jl}(\tilde{x}_l(k)) =$$
$$= c(k)\bar{w}^T(k-1)\bar{\tilde{\psi}}(\tilde{x}(k)) + (1-c(k))\underline{w}^T(k-1)\underline{\tilde{\psi}}(\tilde{x}(k))$$

where

$\underline{w}(k-1) = (\underline{w}_{11}(k-1), \underline{w}_{21}(k-1), \ldots, \underline{w}_{p1}(k-1), \underline{w}_{12}(k-1), \ldots, \underline{w}_{jl}(k-1), \ldots, \underline{w}_{ph}(k-1))^T$,

$\underline{\tilde{\psi}}(\tilde{x}(k)) = (\underline{\tilde{\psi}}_{11}(\tilde{x}(k)), \underline{\tilde{\psi}}_{21}(\tilde{x}(k)), \ldots, \underline{\tilde{\psi}}_{p1}(\tilde{x}(k)), \underline{\tilde{\psi}}_{12}(\tilde{x}(k)), \ldots, \underline{\tilde{\psi}}_{jl}(\tilde{x}(k)), \ldots, \underline{\tilde{\psi}}_{ph}(\tilde{x}(k))^T$,

$\bar{w}(k-1) = \left(\bar{w}_{11}(k-1), \bar{w}_{21}(k-1), \ldots, \bar{w}_{jl}(k-1), \ldots, \bar{w}_{ph}(k-1)\right)^T$,

$\bar{\tilde{\psi}}(\tilde{x}(k)) = \left(\bar{\tilde{\psi}}_{11}(\tilde{x}(k)), \bar{\tilde{\psi}}_{21}(\tilde{x}(k)), \ldots, \bar{\tilde{\psi}}_{jl}(\tilde{x}(k)), \ldots, \bar{\tilde{\psi}}_{ph}(\tilde{x}(k))\right)^T$.

Adaptive Learning of HGAT2FWNN

The learning process of the layer L_1 is the tuning of synaptic weight vectors $\underline{w}(k)$ and $\bar{w}(k)$ of the neurons $\underline{AWN}$, $\overline{AWN}$ and scalar parameter $c(k)$ in the type-reduction block.

Since the output signals of wavelet neurons depend linearly on the synaptic weights, for their settings it is possible to use the exponentially weighted recursive least squares method, which is, in fact, the second-order optimisation procedure

$$\begin{cases} \underline{w}(k) = \underline{w}(k-1) + \dfrac{\underline{P}(k-1)\left(x(k) - \underline{w}^T(k)\underline{\tilde{\psi}}(\tilde{x}(k))\right)}{\beta + \underline{\tilde{\psi}}^T(\tilde{x}(k))\underline{P}(k-1)\underline{\tilde{\psi}}(\tilde{x}(k))}\underline{\tilde{\psi}}(\tilde{x}(k)), \\ \underline{P}(k) = \dfrac{1}{\beta}\left(\underline{P}(k-1) - \dfrac{\underline{P}(k-1)\underline{\tilde{\psi}}(\tilde{x}(k))\underline{\tilde{\psi}}^T(\tilde{x}(k))\underline{P}(k-1)}{\beta + \underline{\tilde{\psi}}^T(\tilde{x}(k))\underline{P}(k-1)\underline{\tilde{\psi}}(\tilde{x}(k))}\right), \end{cases} \tag{15}$$

$$\begin{cases} \bar{w}(k) = \bar{w}(k-1) + \dfrac{\bar{P}(k-1)\left(x(k) - \bar{w}^T(k)\bar{\tilde{\psi}}(\tilde{x}(k))\right)}{\beta + \bar{\tilde{\psi}}^T(\tilde{x}(k))\bar{P}(k-1)\bar{\tilde{\psi}}(\tilde{x}(k))}\bar{\tilde{\psi}}(\tilde{x}(k))), \\ \bar{P}(k) = \dfrac{1}{\beta}\left(\bar{P}(k-1) - \dfrac{\bar{P}(k-1)\bar{\tilde{\psi}}(\tilde{x}(k))\bar{\tilde{\psi}}^T(\tilde{x}(k))\bar{P}(k-1))}{\beta + \bar{\tilde{\psi}}^T(\tilde{x}(k))\bar{P}(k-1)\bar{\tilde{\psi}}(\tilde{x}(k))}\right) \end{cases} \tag{16}$$

where $0 < \beta \leq 1$ – the forgetting factor.

To calculate the parameter $c(k)$ we can introduce an optimal adaptive procedure.

For off-line learning, we can write the learning error in the form

$$e(k)=y(k)-\hat{y}(k)=y(k)-c\overline{y}^{w}(k)-(1-c)\underline{y}^{w}(k)=y(k)-c\overline{y}^{w}(k)-\underline{y}^{w}(k)+c\underline{y}^{w}(k)$$

and the global learning criterion for the type-reduction block in the form

$$\begin{aligned}E(k)&=\sum_{k}\left(x(k)-\hat{x}^{[1]}(k)\right)^{2}=\sum_{k}\left(x(k)-c\overline{y}^{w}-(1-c)\underline{y}^{w}(k)\right)^{2}=\\&=\sum_{k}\left(x(k)-\underline{y}^{w}(k)\right)^{2}+2\sum_{k}\left(c(x(k)-\underline{y}^{w}(k))(\underline{y}^{w}(k)-\overline{y}^{w}(k))\right)+\sum_{k}c^{2}\left(\underline{y}^{w}(k)-\overline{y}^{w}(k)\right)^{2}.\end{aligned} \tag{17}$$

Using criterion (1), we can provide optimization in the form

$$\frac{\partial E(k)}{\partial c}=2\sum_{k}\left((x(k)-\underline{y}^{w}(k))(\underline{y}^{w}(k)-\overline{y}^{w}(k))\right)+2\sum_{k}c\left(\underline{y}^{w}(k)-\overline{y}^{w}(k)\right)^{2}=0 \tag{18}$$

and the expression for calculating parameter c can be written as follows

$$c=-\frac{\sum_{k}\left((x(k)-\underline{y}^{w}(k))(\underline{y}^{w}(k)-\overline{y}^{w}(k))\right)}{\sum_{k}\left(\underline{y}^{w}(k)-\overline{y}^{w}(k)\right)^{2}}. \tag{19}$$

Using the global learning criterion (17), we can obtain the optimal adaptive procedure for tuning parameter c.

Based on (19), we can write the value of parameter c for k and $k+1$ instant time

$$c(k)=-\frac{\sum_{i=1}^{k}\left((x(i)-\underline{y}^{w}(i))(\underline{y}^{w}(i)-\overline{y}^{w}(i))\right)}{\sum_{i=1}^{k}\left(\underline{y}^{w}(i)-\overline{y}^{w}(i)\right)^{2}} \tag{20}$$

or in the recurrent form

$$\begin{cases}\gamma(k)=\gamma(k-1)+\left(\underline{y}^{w}(k)-\overline{y}^{w}(k)\right)^{2},\\ c(k)=c(k-1)\dfrac{\gamma(k-1)}{\gamma(k)}+\dfrac{\left(x(k)-\overline{y}^{w}(k)\right)\overline{y}^{w}(k)\left(\underline{y}^{w}(k)-\overline{y}^{w}(k)\right)}{\gamma(k)}.\end{cases} \tag{21}$$

It is clear that the condition of optimal learning is the closeness of signals $\underline{y}^{w}(k)$ and $\overline{y}^{w}(k)$, “subtending” of type-2 fuzzy wavelet membership function to common “Mexican Hat” wavelet and approximating of parameter $c(k)$ to the value 0.5.

The learning process of layers $L_2, L_3, \ldots, L_g$ is the same. At that after the next observation $x(k)$ is fed, all layers tune their parameters sequentially.

For making a one-step forecast, the signal in the form $X(k+1)=(x(k),x(k-1),\ldots,x(k-n+1))^T$ is fed to the output tuned system, after that the feature observation $\hat{x}^{[g]}(k+1)$ is appeared.

Conclusions

Proposed deep stacking forecasting hybrid system of computational intelligence is enough simple in computational implementation due to parallelizing process of implemented computations, has high learning rate due to dismissal from errors backpropagation and using the rapid adaptive algorithms for tuning its parameters, has also flexibility due to possibility of tuning the activation-membership functions. Proposed system is aimed at solving the wide range Data Stream Mining tasks, such as forecasting, identification, emulation, classification and pattern recognition in online mode. Experiments have confirmed the effectiveness of proposed system under consideration.

Acknowledgement

The paper is published with partial support by the ITHEA ISS (www.ithea.org) and the ADUIS (www.aduis.com.ua)

Bibliography

[Abiyev and Kaynak, 2008] Abiyev R., Kaynak O., Fuzzy wavelet neural networks for identification and control of dynamic plants – a novel structure and a comparative study. IEEE Trans. on Industrial Electronics, vol. 2, no. 55, pp. 3133–3140, 2008. DOI 10.1109/TIE.2008.924018

[Aggarwal, 2007] C. Aggarwal, Data Streams: Models and Algorithms (Advances in Database Systems). Springer, 2007. DOI 10.1007/978-0-387-47534-9

[Alexandridis and Zapranis, 2014] A.K. Alexandridis and A.D. Zapranis, Wavelet Neural Networks: with Applications in Financial Engineering, Chaos, and Classification. John Wiley & Sons, 2014. DOI 10.1002/9781118596272

[Aliev and Guirimov, 2014] R.A. Aliev and B.G. Guirimov, Type-2 Fuzzy Neural Networks and Their Applications. Switzerland: Springer; 2014. DOI 10.1007/978-3-319-09072-6

[Bifet, 2010] A. Bifet, Adaptive Stream Mining: Pattern Learning and Mining from Evolving Data Streams. Amsterdam: IOS Press, 2010. DOI 10.3233/978-1-60750-472-6-i

[Bodyanskiy et all, 2001] Bodyanskiy Ye., Kolodyazhniy V., Stephan A., An adaptive learning algorithm for a neuro fuzzy network. Ed. by B. Reusch “Computational Intelligence Theory and Applications.” Berlin Heidelberg New York: Springer, 2001, pp. 68-75. DOI 10.1007/3-540-45493-4_11

[Bodyanskiy et all, 2005] Bodyanskiy Ye., Kolodyazhniy V., Otto P., Neuro-fuzzy Kolmogorov’s network for time-series prediction and pattern classification. Lecture Notes in Artificial Intelligence. V. 3698. Heidelberg: Springer –Verlag, 2005, pp. 191-202. DOI 10.1007/11551263_16

[Bodyanskiy et all, 2005a] Bodyanskiy Ye., Lamonova N., Pliss I., Vynokurova O., An adaptive learning algorithm for a wavelet neural network. *Expert Systems*, vol. 22, no. 5, pp. 235–240, 2005. DOI 10.1111/j.1468-0394.2005.00314.x

[Bodyanskiy et all, 2008] Bodyanskiy Ye., Popov S., Rybalchenko T., Multulayer neuro-fuzzy network for short term electric load forecasting. Lecture Notes in Computer Science. V.5010. Berlin Heidelberg: Springer-Verlag, 2008, pp. 339-348. DOI 10.1007/978-3-540-79709-8_34

[Bodyanskiy and Vynokurova, 2011] Bodyanskiy Ye., Vynokurova O., Hybrid type-2 wavelet-neuro-fuzzy network for businesses process prediction. Business Informatics, no. 21, pp. 9–21, 2011.

[Bodyanskiy et all, 2012] Bodyanskiy Ye., Pliss I., Vynokurova O., Type-2 Fuzzy-Wavelet-Neuron for Solving Data Mining Problems. In: Proc. East West Fuzzy Colloquium 2012, 19-th Zittau Fuzzy Colloquium. Zittau / Goerlitz: HS, 2012, pp. 96–103.

[Bodyanskiy and Vynokurova, 2013] Bodyanskiy Ye., Vynokurova O., Hybrid adaptive wavelet-neuro-fuzzy system for chaotic time series identification. Information Science, no. 220, pp. 170–179, 2013. DOI 10.1016/j.ins.2012.07.044

[Bodyanskiy et all, 2014] Bodyanskiy Ye., Tyshchenko O., Kopaliani D., A multidimensional cascade neuro-fuzzy system with neuron pool optimization in each cascade. Int. J. of Information Technology and Computer Science. vol.6, iss. 8, pp. 11-17, 2014. DOI 10.1007/s00500-014-1344-3

[Bodyanskiy et all, 2014a] Bodyanskiy Ye., Kharchenko O., Vynokurova O., Least squares support vector machine based on wavelet-neuron. Information Technology and Management Science, vol. 7, pp. 19–24, 2014. DOI 10.1515/itms-2014-0002

[Bodyanskiy et all, 2015] Bodyanskiy Ye., Pliss I.P., Tatarinova Yu., Vynokurova O. Hybrid Generalized Additive Type-2 Fuzzy-Wavelet-Neural Network in Dynamic Data Mining Tasks. Information Technology and Management Science, 18, pp. 70-77, 2015. DOI 10.1515/itms-2015-0011

[Bodyanskiy and Tyshchenko, 2016] Bodyanskiy Ye., Tyshchenko O., Fast Deep Learning for Cascade Neuro-Fuzzy System. In: 2nd Int. Workshop and Networking Event “Advances in Data Science”. Holny Mejera, Poland, 2016, pp. 18-19.

[Bodyanskiy et all, 2016] Bodyanskiy Ye., Mulesa P., Setlak G., Pliss I., Vynokurova O., Fast learning algorithm for deep evolving GMDH-SVM neural network in data stream mining tasks. In: Proc. of

2016 IEEE First International Conference on Data Stream Mining & Processing (DSMP), 2016, pp. 257-262. DOI 10.1109/DSMP.2016.7583555

[Bodyanskiy et all, 2016a] Bodyanskiy Ye., Pliss I., Peleshko D., Vynokurova O., Adaptive Deep Learning of Multivariate Hybrid System of Computational Intelligence. In: VII-th Int. Conf. "Decision-Making Theory", Uzhhorod, Ukraine, 2016, pp. 58-59. (In Ukrainian)

[Du and Swamy, 2014] K.-L. Du, M.N.S. Swamy, Neural Networks and Statistical Learning. London: Springer-Verlag, 2014. DOI 10.1007/978-1-4471-5571-3

[Fahlman and Lebiere, 1990] Fahlman S.E., Lebiere C., The Cascade-Correlation Learning Architecture. Advances Neural Information Processing Systems, San Mateo, CA: Morgan Kaufman, pp. 524-532, 1990. ISBN:1-55860-100-7

[Goodfellow et all, 2016] I. Goodfellow, Y. Bengio, A. Courville. Deep learning. MIT Press, 2016.

[Hastie and Tibshirani, 1990] T. Hastie, R. Tibshirani, Generalized Additive Models. Chapman and Hall/CRC, 1990.

[Haykin, 1999] S. Haykin, Neural Networks. A Comprehensive Foundation. Upper Saddle River. N.J.: Prentice Hall, 1999.

[Huang et all, 2014] Huang W., Song G., Hong H., Xie K. Deep architecture for traffic flow prediction. Deep belief networks with multitask learning. IEEE Trans. Intell. Transp. Syst. vol. 15, is. 5, pp. 2191-2201, 2014. DOI 10.1109/TITS.2014.2311123

[Jang and Mizutani, 1997] J.-S. R. Jang, C. T. Sun, E. Mizutani, Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. N.J.: Prentice Hall, 1997. ISBN:0-13-261066-3

[Längkvist et all, 2014] Längkvist, M., Karlsson, L., Loutfi, A. A review of unsupervised feature learning and deep learning for time-series modeling. Pattern Recognition Letters, vol. 42, is. 1, pp. 11-24, 2014. DOI 10.1016/j.patrec.2014.01.008

[LeCun et all, 2015] LeCun Y., Bengio Y., Hinton G. E., Deep Learning. Nature. 521, pp. 436-444, 2015.

[Mumfold and Jain, 2009] C.L. Mumfold, L.C. Jain, Computational Intelligence Collaboration, Fusion and Emergence. Berlin: Springer-Verlag, 2009. DOI 10.1007/978-3-642-01799-5

[Lughofer, 2011] E. Lughofer, Evolving Fuzzy Systems - Methodologies, Advanced Concepts and Applications. Springer, 2011. DOI 10.1007/978-3-642-18087-3

[Mitaim and Kosko, 1996] Mitaim S., Kosko B., What is the best shape for a fuzzy set in function approximation? In: Proc. 5th IEEE Int. Conf. on Fuzzy Systems "Fuzz-96", vol. 2, 1996, pp. 1237–1213. DOI 10.1109/fuzzy.1996.552354

[Mitaim and Kosko, 1997] Mitaim S., Kosko B., Adaptive joint fuzzy sets for function approximation. In: Proc. Int. Conf. on Neural Networks "ICNN-97", 1997, pp. 537–542. DOI 10.1109/icnn.1997.611726

[Nelles, 2001] O. Nelles, Nonlinear System Identification. Berlin: Springer, 2001. DOI 10.1007/978-3-662-04323-3

[Rutkowski, 2008] L. Rutkowski, Computational Intelligence: Methods and Techniques. Berlin: Springer-Verlag, 2008. DOI 10.1007/978-3-540-76288-1

[Schmidhuber, 2015] Schmidhuber, J., Deep learning in neural networks: An overview. Neural Networks. 61, pp. 85-117, 2015. DOI 10.1016/j.neunet.2014.09.003

[Setlak et all, 2016] Setlak G., Bodyanskiy Ye., Pliss I.P, Vynokurova O., Deep Evolving GMDH-SVM-Neural Network and its Learning for Data Mining Tasks. In: Proc. of the Federated Conference on Computer Science and Information Systems (Annals of Computer Science and Information Systems, Volume 5), 2016. Gda´nsk, Poland, pp. 147–151. Electronic ISBN: 978-8-3608-1090-3

[Sugeno and Kang, 1988] Sugeno M., Kang G.T., Structure identification of fuzzy model. Fuzzy Sets and Systems, vol. 28, pp. 15–33, 1988. DOI 10.1016/0165-0114(88)90113-3

[Takagi and Sugeno, 1985] Takagi T., Sugeno M., Fuzzy identification of systems and its applications to modeling and control. IEEE Trans. on Systems, Man and Cybernetics, vol 15, no. 1, pp. 116–132, 1985. DOI 10.1109/TSMC.1985.6313399

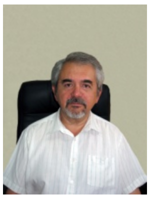
[Takagi and Hayashi, 1991] Takagi H., Hayashi I., NN-driven fuzzy reasoning. Int. J. of Approximate Reasoning, vol. 5, no. 3, pp. 191–212, 1991. DOI 10.1016/0888-613X(91)90008-A

[Uchino and Yamakawa, 1997] Uchino E., Yamakawa T., Soft computing based signal prediction, restoration and filtering. Intelligent Hybrid Systems: Fuzzy Logic, Neural Networks and Genetic Algorithms, Da Ruan Eds., Boston: Kluwer Academic Publisher, pp. 331–349, 1997. DOI 10.1007/978-1-4615-6191-0_14

[Yamakawa et all, 1998] Yamakawa T., A novel nonlinear synapse neuron model guaranteeing a global minimum – Wavelet neuron. In: Proc. 28-th IEEE Int. Symp. on Multiple-Valued Logic. Fukuoka, Japan: IEEE Comp. Soc., 1998, pp. 335-336. DOI 10.1109/ISMVL.1998.679510

[Yamakawa et all, 1992] Yamakawa T., Uchino E., Miki T., Kusanagi H., A neo-fuzzy neuron and its applications to system identification and prediction of the system behaviour. In: *Proc.* 2-nd Int. Conf. on Fuzzy Logic and Neural Networks "IIZUKA-92", Iizuka, Japan, 1992, pp. 477–483.

Authors' Information



Yevgeniy Bodyanskiy *– Dr. Sc., Prof., Control System Research Laboratory, Kharkiv National University of Radio Electronics, 14 Nauky Ave., Kharkiv, 61166, Ukraine;*

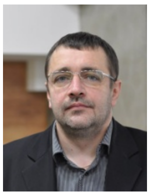
e-mail: yevgeniy.bodyanskiy@nure.ua

Major Fields of Scientific Research: Computational Intelligence, Hybrid Evolving Systems, Data Stream Mining, Clustering, Forecasting.

Iryna Pliss *– Ph.D., Leading reseacher, Control System Research Laboratory, Kharkiv National University of Radio Electronics, 14 Nauky Ave., Kharkiv, 61166, Ukraine;*

e-mail: iryna.pliss@nure.ua

Major Fields of Scientific Research: Data Mining, Computational Intelligence, Hybrid Evolving Systems, Clustering, Forecasting.



Dmytro Peleshko *– Dr. Sc., Prof., University "IT Step Academy", 83a Zamarstynivs'ka st., 79019, Lviv, Ukraine*

e-mail: dpeleshko@gmail.com

Major Fields of Scientific Research: Computational Intelligence, Pattern Recognition, Computer Vision, Image and Video Processing.

Olena Vynokurova *– Dr. Sc., Prof., Control System Research Laboratory, Kharkiv National University of Radio Electronics, 14 Nauky Ave., Kharkiv, 61166, Ukraine;*

e-mail: vynokurova@gmail.com

Major Fields of Scientific Research: Computational Intelligence, Hybrid Wavelet Neural Systems, Data Stream Mining, Clustering, Forecasting.

VIDEO SHOT BOUNDARY DETECTION VIA SEQUENTIAL CLUSTERING

Sergii Mashtalir, Volodymyr Mashtalir, Mykhailo Stolbovyi

Abstract: *To provide condensed and succinct representations of a video stream content there arise huge demand of video parsing. Key frames. shots, scenes, events, scenario represent structural video units. Primary and minimal unit is shot having similar visual contents and can be obtained by video stream clustering. Single-valued features of interest points neighborhoods associated with visual attention regions produce multidimensional time series. Detection of sharp and smooth boundaries of video shot based on fuzzy recurrent clustering is considered.*

Keywords: *I. Computing Methodologies, I.5. Pattern Recognition, I.5.3. Clustering*

Introduction

Large-scale video data collections growth at an exponential rate leads to development of novel on line approaches for structuring of source video streams automatically [Asghar et al, 2014]. Sequential semantic analysis of video structure is a key feature in video summarization, retrieval systems, etc. The peculiarity of video streams processing is on line data acquisition when frames get consistently with rather high frequency. But the foremost challenge when creating a video understanding system remains the semantic gap, i.e. disparity between high-level interpretation of a video and low-level features that can be extracted from frame sequences [Geetha and Narayanan, 2008]. Hierarchical video representation structure is described from the low level to high level: from frames to shots, scenes, events, scenario. The main video parsing units are shots having similar visual contents and can be obtained by video stream clustering. Shot boundary detection in temporal imagery is based usually on low-level frame feature changes such as in descriptions of interest points, boundaries, shapes, colors, textures, object motion, etc. [Kundu and Janwe, 2015]. Small neighborhoods of interest points, which have some identifiable property, have definite promise in video stream partition into elementary uninterrupted content units. In any case frame features may generate a multidimensional time series.

The change detection in properties of multidimensional time series (data sequences) is often encountered in many practical applications [Basseville and Nikiforov, 1993, Badavas, 1993, Pouliezos and and Stavrakakis, 1994]. However, most of the known on-line sequential detection algorithms are oriented to sudden-onset disruptions [Basseville and Nikiforov, 1993], while in video streams, changes can occur fairly smoothly by virtue of various gradual shot changes (dissolve, fade in, fade out, wipe, etc.) or just very slow panning. In such situations, methods of time series fuzzy segmentation [Abonyi et al, 2003, Bodyanskiy and Mashtalir, 2012] are more preferable, however, due to their computational complexity, they become inefficient at high data entry rates for processing. Of course, one can use fuzzy clustering [Badavas, 1993, Aggarwal and Reddy, 2014] and, above all, methods of recurrent clustering [Park and Dagher, 1984, Bodyanskiy, 2005], which, however, suffer from the effects of the ‘curse of dimensionality’ (vectors in high dimensional spaces tend to have roughly the same length) generated by the relatively high dimensionality of the signals being processed. The approach based on the indirect time series clustering (problem is solved by defining a pairwise similarity or dissimilarity measure between series elements) can be very effective in combination with the methods of fuzzy recursive optimization [Hoeppner and Klawonn, 2000]. A technique of on line change properties detection of a multidimensional sequence subject to level clusters intersection is the study object of the paper.

Preprocessing of multidimensional time series for fuzzy clustering

Let original observations be given in the form of multidimensional sequence $x(1),\ldots,x(k),\ldots,x(N)$; $x(k)=(x_1(k),x_2(k),\ldots x_n(k))^T$ where $k=1,2,\ldots,N,\ldots$ denotes current discrete time. According to indirect approach to a time series clustering not the signal $x(k)$, $k=1,2\ldots,N$ is partitioned into clusters The indirect clustering approach solves time series clustering problem by defining a pairwise similarity or by some indirect features such as average, variance, autocorrelation. So, for *i-th* series component, it is possible to introduce recurrent estimates:

– for an average

$$\bar{x}_i(k)=\bar{x}_i(k-1)+\frac{1}{k}\left(x_i(k)-\bar{x}_i(k-1)\right), \tag{1}$$

– for variance

$$\sigma_i^2(k)=\sigma_i^2(k)-\frac{1}{k}\left(x_i(k)-\bar{x}_i(k)^2-\sigma_i^2(k-1)\right)^2, \tag{2}$$

– for autocorrelation coefficients

$$r_i(k,\tau)=r_i(k-1,\tau)+\frac{1}{k}\left(\left(x_i(k)-\bar{x}_i(k)\right)\left(x_i(k-\tau)-\bar{x}_i(k)\right)-r_i(k-1,)\tau\right). \tag{3}$$

Denoting $\bar{x}(k)=(\bar{x}_1(k),\bar{x}_2(k)\ldots\bar{x}_n(k))^T$ and supposing that $\tau=0,1,2,\ldots\tau_{max}$ stands for time lag it is easy enough to identify vector-matrix counterparts of (1) – (3)

$$\bar{x}(k)=\bar{x}(k-1)-\frac{1}{k}\left(x(k)-\bar{x}(k-1)\right), \tag{4}$$

$$R(k,\tau)=R(k-1,\tau)+\frac{1}{k}\Big(\left(x(k)-\bar{x}(k)\right)\left(x(k-\tau)-\bar{x}(k)\right)^{T}-R(k-1,\tau)\Big) \tag{5}$$

where diagonal elements of a symmetric matrix of $R(k,0)$ are, in fact, estimations of variances $r_{ii}(k,0)=\sigma_i^2(k)$ and off-diagonal elements constitute coefficients of mutual correlation $r_{ij}(k,0)$, $i,j=1,2,\ldots n$.

To provide adaptive properties of recurrent procedures (1)-(5), it seems desirable to estimate being studied characteristics on the basis of exponential smoothing modification [Pau, 1981]

$$\bar{x}_i^{\alpha}(k)=\alpha x_i(k)+(1-\alpha)\bar{x}_i^{\alpha}(k-1), \quad 0<\alpha<1, \tag{6}$$

$$(\sigma_i^{\alpha}(k))^2=\alpha\left(x_i(k)+\bar{x}_i^{\alpha}(k)\right)^2+(1-\alpha)\left(\sigma_i^{\alpha}(k-1)\right)^2, \tag{7}$$

$$r_i^{\alpha}(k,\tau)=\alpha\left(x_i(k)-\bar{x}_i^{\alpha}(k)\right)(x_i(k-\tau)-\bar{x}_i^{\alpha}(k))+(1-\alpha)r_i^{\alpha}(k-1,\tau) \tag{8}$$

where $\alpha=2/(L+1)$ is forgetting factor which provides smoothing at the sliding window containing the L last observations $x_i(k),x_i(k-1),\ldots x_i(k-L+1)$.

Thus, $(2+\tau_{max})\times 1$ feature vector $\tilde{x}_i(k)=\left(\bar{x}_i^{\alpha}(k),\ (\sigma_i^{\alpha}(k))^2,\ r_i^{\alpha}(k,1)\ldots r_i^{\alpha}(k,\tau_{max})\right)^T$ can be stated in correspondence to each time series component $x_i(k)$ and namely this vector variety represents clustering objects.

To process $(n+1)$-dimensional signal for an average vector $\bar{x}^{\alpha}(k)$ and an autocorrelation matrix $R^{\alpha}(k,\tau)$ evaluations it is also possible to use the procedure of exponential smoothing in the form [Bodyanskiy et al, 2012]

$$\bar{x}^{\alpha}(k)=A\,x(k)+(I-A)\bar{x}^{\alpha}(k-1) \tag{9}$$

where $A = diag\,(\alpha_1, \alpha_2, \ldots \alpha_n)$, I denotes $(n \times n)$ identity matrix,

$$R_i^{\alpha}(k,) = A\Big(\big(x(k) - \bar{x}^{\alpha}(k)\big)\big(x(k-\tau) - \bar{x}_i^{\alpha}(k)\big)^T\Big)_i + (I - A)R_i^{\alpha}(k) \qquad (10)$$

where index $i = 1, 2, \ldots n$, stands for column number of corresponding matrix.

Formulae (9), (10) application leads to necessity of $0.5\big(n^2 + n\big)(\sigma_{max} + 1) + n$ parameters persistently parsing what considerably complicates real time data processing. In connection with requirements of processing speed increasing, further it is offered for study and, respectively, on-line clustering to use the current values of $((2 + \tau_{max}) \times n)$ matrix $\tilde{x}(\mathrm{k}) = \big(\tilde{x}_1(k), \tilde{x}_2(k), \ldots \tilde{x}_n(k)\big)$ whose elements are refined at each step by means of (6)-(8) relationships.

Change detection of time series components properties

Since changes, occurring during multidimensional signal $\tilde{x}(k)$ processing, can be both jump-like and smooth, fuzzy clustering algorithms (with different fuzzifiers) based on the fuzzy goal functions usage can be applied [Aggarwal and Reddy, 2014].

Now, let formalize a problem. Let an analysis object be the i-th component of the $(\alpha + \tau_{max}) \times 1$ vector signal $\tilde{x}_i(k) = \big(\bar{x}_i^{\alpha}(k),\ (\sigma_i^{\alpha}(k))^2,\ r_i^{\alpha}(k,1) \ldots r_i^{\alpha}(k, \tau_{max})\big)^T$ and the traditional self-learning criterion is used as the objective function

$$E\big(\mu\big(\tilde{x}_i(k), C_i(l)\big)\big) = \sum_{k=1}^{N}\sum_{l=1}^{m}\mu^{\beta}\big(\tilde{x}_i(k), C_i(l)\big)D^2\big(\tilde{x}_i(k), C_i(l)\big) \qquad (11)$$

under constraints

$$\sum_{l=1}^{M}\mu\big(\tilde{x}_i(k), C_i(l)\big) = 1, \quad \forall k = 1, 2, \ldots N, \qquad (12)$$

$$0 < \sum_{k=1}^{N}\mu\big(\tilde{x}_i(k), C_i(l)\big) \le N, \quad \forall l = 1, 2, \ldots, m. \qquad (13)$$

Here, $\mu\left(\tilde{x}_i(k), C_i(l)\right)$ is the membership level of the vector $\tilde{x}_i(k)$ to the l-th cluster, $C_i(l)$ is the centroid of this cluster to be evaluated, β is a nonnegative parameter called the fuzzifier and controlled how much clusters may overlap, $D^2\left(\tilde{x}_i(k), C_i(l)\right)$ denotes some measure specifying the distance between the vectors $\tilde{x}_i(k)$ and cluster $C_i(l)$ in the accepted metric (usually Euclidean in the simplest and most common case). In this case, the objective function (11) can be rewritten in the form

$$E\left(\mu\left(\tilde{x}_i(k), C_i(l)\right)\right) = \sum_{k=1}^{N}\sum_{l=1}^{m}\mu^2\left(\tilde{x}_i(k), C_i(l)\right) \| \tilde{x}_i(k) - C_i(l) \|^2 . \qquad (14)$$

Optimization of objective function (11) by means of the standard nonlinear programming technique leads to the result known as fuzzy C-means clustering procedure

$$\begin{cases} \mu\left(\tilde{x}_i(k), C_i(l)\right) = \dfrac{\| x_i(k) - C_i(l) \|^{-2}}{\sum_{j=1}^{m} \| \tilde{x}_i(k) - C_i(j) \|^{-2}}, \\ C_i(l) = \dfrac{\sum_{k=1}^{N}\mu^2\left(\tilde{x}_i(k), C_i(l)\right)\widetilde{x}_i(k)}{\sum_{k=1}^{N}\mu^2\left(\tilde{x}_i(k), C_i(l)\right)}. \end{cases} \qquad (15)$$

It is easily seen that method (15) describes a batch procedure for information processing, when the entire sample to be analyzed is given and does not change during operations. It is clear that such a procedure can not be used to detect changes under sequential video processing. To achieve specific aims of on line video parsing it will be sufficient to use the adaptive version of (15) [Bodyanskiy, 2005], which allows real-time solving the clustering-segmentation problem as data are fed for processing. In the above notation, the clustering procedure can be written in more promising form

$$\begin{cases} \mu\left(\tilde{x}_i(k), C_i(l,k-1)\right) = \dfrac{\|\tilde{x}_i(k) - C_i(l,k-1)\|^{-2}}{\sum_{j=1}^{m} \|\tilde{x}_i(k) - C_i(j,k-1)\|^{-2}}, \\ C_i(l,k) = C_i(l,k-1) + \eta(k)\,\mu^{\beta}\left(\tilde{x}_i(k), C_i(l,k-1)\right)\left(\tilde{x}_i(k) - C_i(l,k-1)\right) \end{cases} \quad (16)$$

where $0 < \eta(k) < 1$ is the learning rate parameter, chosen usually from empirical considerations.

With $\beta = 2$ procedure (16) will lead to the same results as the batch algorithm (15), but at the same time it allows to process a data sequence that is received in real time for processing. When $\beta = 0$ (16) takes the form of Kohonen's self-learning algorithm [Kohonen, 1995], which is one of the most popular in solving clustering problems. Note also that conditions $\beta = 0$, $\eta(k) = k^{-1}$ correspond to the recurrent version of the k-means clustering [Aggarwal and Reddy, 2014], widely used in video and image processing. Finally, an important advantage (17) is that preconditions are created for filtering some local distortions, in particular, occlusions.

Change properties detection of a multidimensional sequence

When segmenting video streams with sequential mode of one-dimensional discrepancies detection, an application of usual pixels as a time series generator is not valid. Explanation is simple: in the frame sequence for such processing it is possible to use only the gray level (color) values at given coordinates. In other words, the spatial content of the image is completely excluded from consideration that increases the semantic gap. In addition, the impact of various disturbing influences, primarily distorting the coordinates, hides natural temporal dynamics of the point to be analyzed. Thus, it is necessary to consider point neighborhoods properties producing time series. For sequential video clustering in order to reduce the semantic gap, it is most expedient to select, as point-generators of time series, interest points which are sufficiently uniformly distributed in the field of view and preferably are inside visual attention regions what further increases the interpretability of the detected discrepancies.

Interest points (corners, contour junctions, ridges extremum, etc.) utilizing may provide feature point correspondences between frames from video stream and, ipso facto, generate multidimensional time series corresponding to video content changes with high degree of robustness since they have distinctness, well-defined position, local information contents, are stable under illumination/brightness variations. Thus, shot boundary detection may be based on single-valued textures or another features defined in neighborhoods of various interest points.

Since in the detection process described above, n one-dimensional type (16) procedures are simultaneously realized, from the computational point of view it is more efficient to process at once parameters of the entire matrix $\tilde{x}(k)$ with dimensionality $((2+\tau_{max})\times n)$. To do this, it is advisable to use the fuzzy C-means technique modification where instead of objective function (14) it can be exploited expression

$$E(\mu(\tilde{x}(k),C(l))=\sum_{k=1}^{N}\sum_{l=1}^{m}\mu^2(\tilde{x}(k),C(l))Sp(\tilde{x}(k)-C(l))(\tilde{x}(k)-C(l))^T. \quad (17)$$

Here Frobenius norm metric is applied instead of the Euclidean one and, in addition, m matrix-centroids $C(l)$ are introduced.

Minimization of (17) with constraints (12), (13) leads to analytical expression

$$\begin{cases} \mu(\tilde{x}(k),C(l))=\dfrac{\left(Sp(\tilde{x}(k)-C(l))(\tilde{x}(k)-C(l))^T\right)^{-1}}{\sum_{j=1}^{m}\left(Sp(\tilde{x}(k)-C(j))(\tilde{x}(k)-C(j))^T\right)^{-1}}, \\ C(l)=\dfrac{\sum_{k=1}^{N}\mu^2(\tilde{x}(k),C(l))\tilde{x}(k)}{\sum_{k=1}^{N}\mu^2(\tilde{x}(k),C(l))}. \end{cases} \quad (18)$$

Obviously, relationships (18) can not be utilized to solve on-line being studied problem. In this case, it is expedient to use the matrix modification of the recurrent procedure (16), which in this case has the form

$$\begin{cases} \mu(\tilde{x}(k),C(l,k-1))=\dfrac{\left(Sp(\tilde{x}(k)-C(l,k-1))(\tilde{x}(k)-C(l,k-1))^T\right)^{-1}}{\sum_{j=1}^{m}\left(Sp(\tilde{x}(k)-C(j,k-1))(\tilde{x}(k)-C(j,k-1))^T\right)^{-1}}, \\ C(l,k)=C(l,k-1)+\eta(k)\mu^{\beta}(\tilde{x}(k),C(l,k-1))(\tilde{x}(k)-C(l,k-1)) \end{cases} \quad (19)$$

where the learning rate parameter $\eta(k)$ is chosen from the same considerations as in (16).

It can also be noted that when $\beta = 0$ (19) takes the form of a clear matrix self-learning algorithm for on-line clustering problem solutions as follows

$$C(l,k) = C(l,k-1) + \eta(k)\left(\tilde{x}(k) - C(l,k-1)\right)$$

which is, in fact, a matrix version of the of self-learning rule of T. Kohonen principle ‘Winner Takes All’.

Conclusion

Any real problem of video understanding is extremely difficult due to necessity of valid on line parsing of enormous image sequences in feature or signal spaces. Though video shot boundary detection techniques have achieved significant progress, features to be utilized and reflect desirable semantic level remain an open issue. One from promising approaches is to use interest points associated with visual attention regions, more precisely, the local neighborhoods of these interest points. In this case semantic gap may be decreased and sufficiently high-speed recurrent procedures can be achieved. The problem of change detection in multidimensional (matrix) data streams getting for processing in real time is considered. The on-line procedures for fuzzy clustering have been introduced. It makes possible on line clustering for both slow and abrupt changes. The proposed approach is fairly simple and can find different application, first of all, when processing video streams.

Acknowledgment

The paper is published with partial support by the ITHEA ISS (www.ithea.org) and the ADUIS (www.aduis.com.ua)

Bibliography

[Abonyi et al, 2003] Abonyi J., Feil B., Nemett S., Arva P., Fuzzy clustering based segmentation of time-series. Lecture Notes in Computer Science. 2810. Berlin: Springer, 2003. pp. 275–285. ISBN 978-3-540-40813-0 (print version), ISBN 978-3-540-45231-7 (on-line), DOI: 10.1007/978-3-540-45231-7_26 http://link.springer.com/chapter/10.1007/978-3-540-45231-7_26.

[Aggarwal and Reddy, 2014] Aggarwal C.C., Reddy C.K. Data clustering: Algorithms and applications. Boca Raton: CRC Press, 2014.

[Asghar et al, 2014] Asghar M.N., Hussain F., Manton R., Video indexing: a survey. International Journal of Computer and Information Technology, Vol. 3, Issue 1. 2014. pp. 148-169. ISSN: 2279-0764, http://www.ijcit.com/archives/volume3/issue1/Paper030123.pdf

[Badavas, 1993] Badavas P.C. Real-time statistical process control. Englewood Cliffs: Prentice-Hall, 1993.

[Basseville and Nikiforov, 1993] Basseville M., Nikiforov I. Detection of abrupt changes: Theory and application. Englewood Cliffs, N.J.: PTR Prentice-Hall, 1993.

[Bodyanskiy and Mashtalir, 2012] Bodyanskiy Ye.V., Mashtalir S.V., Search for video stream changes via multidimensional time series analysis. Reports of the National Academy of Sciences of Ukraine, No11, 2012. pp. 30-33, ISSN 1025-6415 (print version), ISSN 2518-153X (on-line), https://dopovidi.nas.gov.ua/2012-11/12-11-05.pdf

[Bodyanskiy et al, 2012] Bodyanskiy Ye., Kinoshenko, D., Mashtalir, S., Mikhnova, O., On-line video segmentation using methods of fault detection in multidimensional time sequences. International Journal of Electronic Commerce Studies, Vol. 3, Issue 1. 2012. pp. 1-20. ISSN 2073-9729, http://academic-pub.org/ojs/index.php/ijecs/article/download/1010/122&authuser=1&usg=AFQjCNEhzFr7iL5POexC5yhRmCiP -pVeQg&sig2=tAh3O9zVY5KXPk5fNgPC-g&bvm=bv.150729734,d.bGg.

[Bodyanskiy, 2005] Bodyanskiy Ye., Computational intelligence techniques for data analysis. Lecture Notes in Informatics, Vol. 72. Bonn: GI, 2005. pp. 15-36. ISBN 3-88579-401-2 http://subs.emis.de/LNI/Proceedings/Proceedings72/GI-Proceedings.72-1.pdf

[Geetha and Narayanan, 2008] Geetha P.A., Narayanan V., Survey of content-based video retrieval. Journal of Computer Science, Vol. 4, Issue 6. 2008. pp. 474-486. ISSN 1549-3636, DOI: 10.3844/jcssp.2008.474.486, http://thescipub.com/PDF/jcssp.2008.474.486.pdf.

[Hoeppner and Klawonn, 2000] Hoeppner, F., Klawonn, F., Fuzzy clustering of sampled functions. In: Proc. 19th Int. Conf. of the North American Fuzzy Information Processing Society (NAFIPS), Atlanta, USA. 2000. pp. 251 – 255. ISBN: 0-7803-6274-8 (print version), DOI: 10.1109/NAFIPS.2000.877431, http://public.fh-wolfenbuettel.de/~hoeppnef/paper/Hoeppner-NAFIPS-2000.pdf

[Kohonen, 1995] Kohonen, T. Self-Organizing Maps. Berlin: Springer-Verlag, 1995.

[Kundu and Janwe, 2015] Kundu A., Janwe N., A survey on video segmentation the future roadmap. International Journal of Modern Trends in Engineering and Research, Vol. 2, Issue 3. 2015. pp. 527-534. ISSN: 2349-9745 (on-line), ISSN: 2393-8161 (print version), http://www.academia.edu/13545611/ a_survey_on_video_segmentation_the_future_roadmap

[Park and Dagher, 1984] Park D.C., Dagher I., Gradient based fuzzy c-means (GBFCM) algorithm. In: IEEE Int. Conf. on Neural Networks, 1984. pp. 16.26-16.31. ISBN: 0-7803-1901-X (print version), DOI: 10.1109/ICNN.1994.374399, http://ieeexplore.ieee.org/document/374399/

[Pau, 1981] Pau L.F. Failure Diagnosis and performance monitoring. NY: Marcel Dekker Inc., 1981.

[Pouliezos and and Stavrakakis, 1994] Pouliezos A.D., Stavrakakis G.S. Real-time fault monitoring of industrial processes. Dordrecht: Kluwer Academic Publishers, 1994.

Authors' Information

Mashtalir Sergii *– Kharkiv National University of Radio Electronics, ass. professor of Informatics department. 14, Nauky Ave., Kharkiv, 61166, Ukraine;*
e-mail: sergii.mashtalir@nure.ua

Major Fields of Scientific Research: Temporal video streams processing, image processing

Mashtalir Volodymyr *– Kharkiv National University of Radio Electronics, professor of Informatics department. 14, Nauky Ave., Kharkiv, 61166, Ukraine;*
e-mail: volodymyri.mashtalir@nure.ua

Major Fields of Scientific Research: Pattern recognition, image processing

Stolbovyi Mykhailo *– Kharkiv National University of Radio Electronics, PhD student. 14, Nauky Ave., Kharkiv, 61166, Ukraine*

Major Fields of Scientific Research: Pattern recognition, image and video processing

AN APPROACH FOR DESIGN OF ARCHITECTURAL SOLUTIONS BASED ON SOFTWARE MODEL-TO-MODEL TRANSFORMATION

Elena Chebanyuk, Kyryl Shestakov

***Abstract**: Software models are central software development artifacts in Model-Driven Software Engineering (MDSE) approach. During the Requirements Analysis, Software Design and Software Development a variety of software models, represented as UML diagrams, are created. However, in a majority of software development approaches they are designed mainly. Considering the fact that new software models contain information from previous ones, software Model-to-Model transformation techniques facilitate the process of new software models obtaining by means of reusing information from software models created before.*

An approach for performing Model-to-Model transformation, which is based on graph transformation, is presented in this paper. Transformational operations are considered on meta-level and concrete level. Software models are represented as graphs.

Software tool, that allows performing transformation from communication diagram to class one, is described. This tool is designed as a plug-in for IBM Rational Software Architect (RSA) 7.5.5.2. Engine medini QVT, integrated in IBM RSA project, supports execution of QVT-R script. Source software models (communication diagrams) are designed in IBM Rational Software Architect. Target software models are also processed (opened) by means of IBM RSA.

***Keywords**: Model-Driven Software Engineering, Model-Driven Development, Model-to-Model Transformation, Graph Transformation, Transformation Rule, IBM Rational Software Architect, medini QVT, Requirement Analysis, Software Designing, UML Diagram, UML Communication Diagram, UML Class Diagram.*

***ACM Keywords**:*

***- Classification of 2012 year**: Software and its engineering, Software system structure, Software system model, Model-Driven system engineering.*

***- Classification of 1998 year:** D2 software engineering, D 2.0 Tools*

Introduction

Software modeling is an integral activity that is performed on software project development that follows a Model-Driven Development (MDD) approach. It is applied to express software with the aim of reducing risk and managing complexity of activities, performed in different software development processes. And the challenge of projects that heavily rely on modeling is to transform designed software models. A software model transformation, which is an automated way of processing models, can be written in a general-purpose programming language, such as Java. However, using a special-purpose transformation language is a more flexible approach since it does not require complex and redundant coding constructs but focuses on relevant domain features. Also using a special-purpose model transformation language increases maintainability, because changes introduced into transformation scripts require less effort to be implemented and no recompilation since scripts are not embedded into transformation software.

Previously, an approach to transform models into text was used exclusively because a final goal of the transformation was to obtain code for the development of software. Plus, the transformation was fixed and completely not capable of change in the implementation of a transformation tool. With the introduction of Model-to-Model transformation activity the objective did not change but rather an additional layer of abstraction was included in order to increase the quality of desired result. This is accomplished by taking advantage of existing semantic relations between different representations of a model to be developed.

The techniques employed for modeling are standardized by the Object Management Group (OMG). Among them are Unified Modeling Language (UML) to define software models expressed as diagrams, Object Constraint Language (OCL) to specify constraints to objects, etc. These widely-accepted techniques are often used to define software requirements in an Agile approach when interacting with a customer. After the meetings with a customer a great number of diagrams is created through which a model is generated. This serves as a substantial input for the process of Model-to-Model transformation. The key in the proposed approach is to transform into a model that reflects software architecture better than other representations. So Model-to-Model transformation fits mostly between the stages of software analysis and design.

In order to implement Model-to-Model software transformation process on a project that follows MDD approach the hardest part is to figure out appropriate relationships between the model elements of given and desired model views. This might make a lot of time analyzing, designing and developing software to gain insight on the relevant model patterns that can be mapped into each other under conditions that should hold. But according to Model-to-Model transformation approach, this kind of experience helps automate the process of software design, which reduces time and effort needed to develop software.

Instead of creating necessary design all the time wasting effort that could be invested into other pressing project matters, required software model can be generated from existing model.

This approach is customary to use at large projects that develop software with high level of complexity, because only such types of software can provide a necessary amount of models for the discovery and analysis of transformation rules. And due to their size, they can be primary beneficiaries of considered approach for saving time and effort, which are valuable resources that have direct correlation with money.

Review of software Model-to-Model transformation types

Model transformations and languages for them have been classified in many ways. Some of the more common distinctions drawn are:

Number and type of inputs and outputs:

In principle a model transformation may have many inputs and outputs of various types; the only absolute limitation is that a model transformation will take at least one model as input. However, a model transformation that did not produce any model as output would more commonly be called a model analysis or model query.

Endogenous versus exogenous:

Endogenous transformations are transformations between models expressed in the same language. Exogenous transformations are transformations between models expressed using different languages. For example, in a process conforming to the OMG Model-Driven Architecture, a platform-independent model might be transformed into a platform-specific model by an exogenous model transformation.

Unidirectional versus bidirectional:

A unidirectional model transformation has only one mode of execution: that is, it always takes the same type of input and produces the same type of output. Unidirectional model transformations are useful in compilation-like situations, where any output model is read-only. The relevant notion of consistency is then very simple: the input model is consistent with the model that the transformation would produce as output, only.

For a bidirectional model transformation, the same type of model can sometimes be input and other times be output. Bidirectional transformations are necessary in situations where people are working on more than one model and the models must be kept consistent. Then a change to either model might necessitate a change to the other, in order to maintain consistency between the models. Because each model can incorporate information which is not reflected in the other, there may be many models which are consistent with a given model.

Horizontal vs Vertical Transformation:

A horizontal transformation is a transformation where the source and target models reside at the same abstraction level (e.g. Platform independent or platform specific levels). Typical examples are refactoring (an endogenous transformation) and migration (an exogenous transformation). A vertical transformation is a transformation where the source and target models reside at different abstraction levels. A typical example is refinement, where a specification is gradually refined into a full-fledged implementation, by means of successive refinement steps that add more concrete details.

Syntactic versus semantic transformations:

A final distinction can be made between model transformations that merely transform the syntax, and more sophisticated transformations that also take the semantics of the model into account. As an example of syntactical transformation, consider a parser that transforms the concrete syntax of a program (resp. model) in some programming (resp. modeling language) into an abstract syntax. The abstract syntax is then used as the internal representation of the program (resp. model) on which more complex semantic transformations (e.g. refactoring or optimization) can be applied. Also when we want to import or export our models in a specific format, a syntactical transformation is needed [Errata et al, 2015].

Model Transformation Languages and Tools

Atlas Transformation Language

Atlas Transformation Language (ATL) is a model transformation language and toolkit developed and maintained by OBEO and INRIA-AtlanMod. It was initiated by the AtlanMod team (previously called ATLAS Group). In the field of Model-Driven Engineering, ATL provides ways to produce a set of target models from a set of source models. Released under the terms of the Eclipse Public License, ATL is an Model-to-Model (Eclipse) component, inside of the Eclipse Modelling Project.

ATL is based on the QVT which is an Object Management Group standard for performing model transformations. It can be used to do syntactic or semantic translation. ATL is built on top of a model transformation Virtual Machine [Errata et al, 2015].

Janus Transformation Language

Janus Transformation Language (JTL) is a bidirectional model transformation language specifically designed to support non-bijective transformations and change propagation. In Model-Driven Engineering, bidirectional transformations are considered as core ingredients for managing both the consistency and synchronization of two or more related models. However, while non-bijectivity in bidirectional transformations is considered relevant, most of the languages lack of a common understanding of its semantic implications hampering their applicability in practice.

The JTL is a bidirectional model transformation language specifically designed to support non-bijective transformations and change propagation. In particular, the language propagates changes occurring in a model to one or more related models according to the specified transformation regardless of the transformation direction. Additionally, whenever manual modifications let a model be non-reachable anymore by a transformation, the closest model which approximate the ideal source one is inferred. The language semantics is also presented and its expressivity and applicability are validated against a reference benchmark. JTL is embedded in a framework available on the Eclipse platform which aims to facilitate the use of the approach, especially in the definition of model transformations [Cicchetti et al, 2010].

Epsilon family is a model management platform that provides transformation languages for model-to-model, model-to-text, update-in-place, migration and model merging transformations. Epsilon Transformation Language (ETL) is a hybrid, rule-based Model-to-Model transformation language. ETL provides all the standard features of a transformation language but also provides enhanced flexibility as it can transform many input to many output models, and can query/navigate/modify both source and target models.

Although a number of successful model transformation languages have been currently proposed, the majority of them have been developed in isolation and as a result, they face consistency and integration difficulties with languages that support other model management tasks. ETL, a hybrid model transformation language that has been developed atop the infrastructure provided by the Epsilon model management platform. By building atop Epsilon, ETL is seamlessly integrated with a number of other task specific languages to help to realize composite model management workflows [Errata et al, 2015].

Kermela

The Kermeta language was initiated by Franck Fleurey in 2005 within the Triskell team of IRISA. The Kermeta language borrows concepts from languages such as MOF, OCL and QVT, but also from BasicMTL, a model transformation language implemented in 2004 in the Triskell team by D. Vojtisek and F. Fondement. It is also inspired by the previous experience on MTL, the first transformation language created by Triskell, and by the Xion action language for UML. Kermeta, and its execution platform are available under Eclipse. It is open-source, under the EPL.

Kermeta is a general purpose modelling and programming language for metamodel engineering which is also able to perform model transformations. It fills the gap of MOF which defines only the structure of meta-models, by adding a way to specify static semantic (similar to OCL) and dynamic semantic (using operational semantic in the operation of the metamodel). Kermeta uses the object-oriented paradigm like Java or Eiffel.

Kermeta is a modeling and aspect oriented programming language. Its underlying metamodel conforms to the EMOF standard. It is designed to write programs which are also models, to write transformations of models (programs that transform a model into another), to write constraints on these models, and to execute them). The goal of this model approach is to bring an additional level of abstraction on top of the "object" level and thus to see a given system like a set of concepts (and instances of concepts) that form an explicitly coherent whole, which one will call a model [Errata et al, 2015].

Querry/View/Transform language

The OMG has defined a standard for expressing M2M transformations, called MOF/QVT or in short QVT [Eclipse, 2016]. Eclipse has two extensions for QVT called QVTd (Declarative) and QVTo (Operational/Procedural). QVT Operational component is a partial implementation of the Operational Mappings Language defined by the OMG standard specification (MOF) 2.0 Query/View/Transformation. In long term, it aims to provide a complete implementation of the operational part of the standard . A high level overview of the QVT Operational language is available as a presentation from EclipseCon 2008, Model Transformation with Operational QVT [Macedo and Cunha, 2013].

AToM3 is a Python based tool for multi-paradigm modeling which stands for "A Tool for Multi-formalism and Meta-Modelling"'. The two main tasks of AToM3 are meta-modeling and model-transforming. Meta-modelling refers to the description, or modelling of different kinds of formalisms used to model systems (although we have focused on formalisms for simulation of dynamical systems, AToM3's capabilities are not restricted to these.) Model-transforming refers to the (automatic) process of converting, translating or modifying a model in a given formalism, into another model that might or might not be in the same formalism.

In AToM3, formalisms and models are described as graphs. From a meta-specification of a formalism, AToM3 generates a tool to visually manipulate (create and edit) models described in the specified formalism. Model transformations are performed by graph rewriting. The transformations themselves can thus be declaratively expressed as graph-grammar models.

Some of the meta-models currently available are: Entity-Relationship, Deterministic Finite State Automata, Non-Deterministic Finite State Automata, Petri Nets, Data Flow Diagrams and Structure Charts. Typical model transformations include model simplification (e.g., state reduction in Finite State Automata), code generation, generation of executable simulators based on the operational semantics of formalisms, as well as behavior-preserving transformations between models in different formalisms. Atom3 is supported by a web based tool, but it has no standalone framework or any integration with a framework such as Eclipse [Lara and Vangheluwe, 2002].

Also, there are other model transformation languages and tools which are mostly under-research and academic studies. Some of them are listed below:

HOTs: Just as any other model can be created, modified, augmented by a transformation, a transformation model can itself be instantiated, modified and so on, by a so-called HOT. This uniformity has several benefits: especially it allows reusing tools and methods, and it creates a framework that can be applied recursively (since transformations of transformations can be transformed themselves) [Massito et al, 2009].

GReAT: It is a transformation language in the generic modeling environment (GME). GReAT language is a graphical language for the specification of graph transformations between Domain-Specific Modelling Languages (DSMLs). It consists of three sub-languages:

– the pattern specification language;
– the transformation rule language;
– the sequencing or control flow language.

Additionally, the input and the output languages of a transformation are defined in terms of meta-models. GReAT is not a standalone tool; rather, it is used in conjunction with the Generic Modelling Environment. However, once a transformation has been developed, a standalone executable can be executed outside of GME [Balasubramanian et al, 2007].

Very detailed review of model transformation approaches and technologies is represented in paper. Authors consider all Model-to-Model transformation mentioned about and a number of other languages, namely Henshin,

MOLA, SiTra, Stratego/XT,Tefkat,Tom,UML-RSDS, and VIATRA2 [Errata et al, 2015].

Related Papers

Papers, related to software model transformation, can be divided to two classes, namely those which make strong contribution in transformation techniques and those that develop analytical tools for designing new and improving existing transformational approaches and techniques.

Detail review of papers, devoting to designing transformation methods and techniques grounded on practical tools and environments is represented in paper [Chebanyuk and Markov, 2016a]. The result of this review is summarizing achievements of researches according MDE promising. List of MDE promising is also represented in paper [Chebanyuk and Markov, 2016a]. Analyzing this review, the requirements to analytical automated method for Model-to-Model transformations, that cover all MDE promising, were formulated.

Also represent review of papers, making strong contribution in development of transformational techniques.

Paper [Greiner et al, 2016] represents a case study dealing with incremental round-trip engineering of UML class models and Java source code.

Described approach tries to prevent information loss during round-trip engineering by using a so called trace model which is used to synchronize the Platform Independent and the Platform Specific Models. Furthermore, the source code is updated using a fine grained bidirectional incremental merge. Also, information loss is prevented by using Javadoc tags as annotations. Case model and code are changed simultaneously and the changes are contradicting, one transformation direction has to be chosen, which causes that some changes might get lost [Greiner et al, 2016].

The contribution of the survey [Seifermann and Groenda, 2016] is the identification and classification of textual UML modeling notations. During the survey, authors found a total of 31 textual UML notations. The classification is aimed to include the user’s point of view and support the notation selection in teams. In total, authors found 14 new notations compared to previous surveys: Alf, Alloy, AUML, Clafer, Dcharts, HUTN, IOM/T, Nomnoml, pgf-umlcd, pgf-umlsd, tUML, txtUML, UML/P, and uml-sequence-diagram-dsl-txl. Authors presented each of the twenty categories in detail including objectively checkable conditions that cover the level of UML support, the editing experience, and the applicability in an engineering team.

Paper [Wu, 2016] addresses the issue of generating metamodel instances satisfying coverage criteria.

More specifically, this paper makes the following contributions:

- – A technique that enables metamodel instances to be generated so that they satisfy partition-based coverage criteria.
- – A technique for generating metamodel instances which satisfy graph properties.

A metamodel is a structural diagram and can be depicted using the UML class diagram notation. Thus, the coverage criteria defined for UML class diagram can also be borrowed for metamodels. To facilitate the transformation from class diagrams with OCL constraints to Satisfiability Modulo Theories (SMT) formulas authors use a bounded typed graph as an intermediate representation [Wu, 2016].

Paper [Natschlager et al, 2016] presents concept for Adaptive Variant Modeling (AdaVM). AdaVM is a part of the AdaBPM framework for advanced adaptability and exception handling in formal business process models. In addition, AdaVM considers linking of elements, propagation of changes, and visual highlighting of differences in process variants. Authors showed that graph transformation techniques are well-suited for process variant management and that variants can be automatically created by a few graph transformation rules specifying concrete variations. Authors show that the adaptable approach is less complex regarding the type graph, source graphs, and the number of rules and application conditions. New ideas, expressed in proposed approaches, are (i) the support of variability by restriction and by extension with graph transformation techniques, (ii) linking and propagation of changes, (iii) individual blocking of elements/attributes, and (iv) visual highlighting of differences in process variants.

A review of mathematical foundations for providing realization of model transformation techniques is outlined in [Rabbi et al, 2016].

A review of metamodeling tools is represented in paper [Favre and Duarte, 2016] and several metamodeling frameworks are described.

Task and Challenges

To propose an approach and software tool for Model-to-Model transformation.

Challenges: the proposed approach should:

- be based on graph transformations to support easy modifying of transformation rules;
- easily allow modifying transformation rules;
- represent of a transformation rules from behavioral software models to class diagrams at metalevel;

Designed software tool should automate the process of Model-to-Model transformation in IBM Rational Software Architect 7.5.5.2;

Proposed Approach

Step 1: Define transformation rules. These transformation rules serve as instructions to a transformation engine on how to transform a model. They reflect the mapping patterns between source and target models elements. A developer has to identify which elements of a source model correspond to which elements of a target model.

Step 2:Set up an execution environment. Eclipse plugin with QVT-R script is integrated to IBM Rational Software Architect (IBM RSA). IBM RSA version, used in this paper is 7.5.5.2 and is based on Eclipse 3.4 Ganymede. The tool to be used for Model-to-Model transformation is medini QVT. Medini QVT is a tool set for Model-to-Model transformations.

Step 3: Create a transformation script in medini QVT. The transformation rules are defined using QVT-R language in medini QVT environment. QVT-R script consists of transformations, but it is customary to have a single transformation defined per one script. A transformation, in turn, considers a set of relations between elements of source and target software models. These relations define the patterns that are used to match model elements. Each relation touches of source and target domains. They specify model elements that are to be mapped into each other by binding their properties to variables that can be used in OCL expressions. Additionally, relations can have where and when clauses that specify conditions under which a relation holds.

Step 4: Design source model in RSA environment. RSA is a great tool that provides modeling capabilities for a developer to create models. In a project considered in the previous step, create a model. The best way to define elements of this model is to build a diagram. Create a necessary diagram and open a designer. Whenever you drag the notation elements onto the drawing surface the corresponding model elements are generated in a model container.

Step 5: Execute the transformation. Now you can click on the command the launches a designed plugin. A “Transformation Definition” dialog appears. In a “QVT-R script” field type a path to a QVT-R script or use the “Browse…” button to choose it using File Explorer. In an “Input model” field type a path to a file containing the source model created in a previous step, or use the “Browse…” button to choose it using File Explorer. After that click “OK” button to launch a transformation. As a result, a target model file is created in the same folder as the source model file.

Description of Transformation Rules

A model transformation, in model-driven engineering, is an automated way of modifying and creating models. An example use of model transformation is ensuring that a family of models is consistent, in a precise sense which the software engineer can define. The aim of using a model transformation is to save effort and reduce errors by automating the building and modification of models where possible.

Model transformations can be thought of as programs that take models as input. There is a wide variety of kinds of model transformation and uses of them, which differ in their inputs and outputs and also in the way they are expressed [Kuster, 2011].

A model transformation usually specifies which models are acceptable as input, and if appropriate what models it may produce as output, by specifying the metamodel to which a model must conform [Rebull et al, 2007].

Consider UML diagram or software model as a graph, where UML diagram objects are graph vertices and UML diagram links are edges of this graph. Thus, consider an analytical background for representing transformation rules. Denote software model(SM) as: SM(O,L), where

O – a set of software model objects. Objects are elements of software model (SM) notations that can be expressed as graph vertexes.

L – a set of links between O, that can be expressed as graph edges. Links are elements of software model notation that can be expressed as edges [Wiemann, 1995].

As in transformation operation there are two software models define them as initial ($SM_{initial}$) and resulting ($SM_{resulting}$). $SM_{initial}$ is a software model from which transformation is started. This model contains initial information for transformation. $SM_{resulting}$ is the model which is obtained after transformation. Thus:

$$\begin{aligned}&SM_{initial} = (O_1, L_1); O_1 = \{o_{1,i} \mid i = 1,...,n_1\};\\&L_1 = \{l_{1,j} \mid j = 1,...,m_1\};\ n_1 = |O_1|;\ m_1 = |L_1|\\&SM_{resulting} = (O_2, L_2); O_1 = \{o_{2,k} \mid k = 1,...,n_2\}\\&L_2 = \{l_{2,p} \mid p = 1,...,m_2\};\ n_2 = |O_2|;\ m_2 = |L_2|\end{aligned} \quad (1)$$

where O_1 - set of $SM_{initial}$ objects, O_2 - set of $SM_{resulting}$ objects.

L_1 - set of $SM_{initial}$ links. L_2 - set of $SM_{resulting}$ links.

Initial and *resulting* are types of software models. For example if transformation performed from use case to communication diagram we write transform SM_{use_case} to $SM_{collaboration}$.

To represent transformation rules transformational grammar [Chomsky, 1957] is used. Transformation rules are syntax of this grammar [Chebanyuk and Markov, 2016a].

Denote transformation operation as “ $\rightarrow$ ” [Chomsky, 1957]. Thus, transformation operation from SMI_{sub} to SMR_{sub} is written as follows:

$$SM_{initial} \rightarrow SM_{resulting} \quad (2)$$

Expression (5) means that applying set of transformation rules, sub-graphs, formed from initial software models are transformed to sub-graphs of resulting software model.

Considering graphs as a set of objects and links, the next expression can be written:

$$\begin{aligned}&(OI, LI) \rightarrow (OR, LR);\\&OI \subset O_1,\ OR \subset O_2,\ LI \subset L_1,\ LR \subset L_2\end{aligned} \quad (3)$$

Represent denotation for Behavioral Software Model (BSM). they were introduced in paper [Chebanyuk, 2015]. Table 1 illustrates denotation for different BSM object.

XMI representation, based on abstract syntax tree concept, allows representation all UML diagrams in unified format.

Using this fact let us formulate one-to-one transformation rule for all elements of different UML diagram notation, that mean the same role of business processes. ***UML diagrams objects that mean the same and have different names in different UML diagrams are transformed from source UML diagram to target one by means of mapping one-to-one.***

Table 2 shows that when objects play the same role in different BSM they are transformed one-to-one. They could be named the same (for example Y_C is transformed to Y_S) or having different names (for example P_{UC} is transformed to M_S).

Table 1. Denotations of BSM objects

Denotation from the UML standard	Denotation of the proposed approach	Use case	Collaboration	Sequence
1	2	3	4	5
Object	O	-	+	+
1	2	3	4	5
Complex object (object with properties defined)	O_{COM}	-	+	-
Actor	A	+	+	+
Message	M	+ (precedent is an equivalent of message)	+	+
Collection	Y	-	+	+
Subsystem	SS	+	+	+
Multiplicity	N	+	+	-
Waiting for response	I	-	-	+

Table 2. One-to-one Model-to-Model transformation mappings

Denotation from the UML standard	Denotation of the proposed approach	Use case (UC)	Collaboration ©	Sequence (S)
1	2	3	4	5
Object	O	-	$O_C \to O_S$	$O_S \to O_C$
Complex object (object with properties defined)	O_{COM}	-	$O_{COM} \to O_s$	-
Actor	A	$A_{UC} \to A_C$ $A_{UC} \to A_S$	$A_C \to A_{UC}$ $A_C \to A_S$	$A_S \to A_{UC}$ $A_S \to A_C$
Message (precedent)	M (P)	$P_{UC} \to M_C$ $P_{UC} \to M_S$	$M_C \to P_{UC}$ $M_C \to M_S$	$M_S \to P_{UC}$ $M_S \to M_C$
Collection	Y	-	$Y_C \to Y_S$	$Y_S \to Y_C$
Subsystem	SS	$SS_{UC} \to SS_C$ $SS_{UC} \to SS_S$	$SS_C \to SS_{UC}$ $SS_C \to SS_S$	$SS_S \to SS_{UC}$ $SS_S \to SS_C$

Setting up execution environment

The primary tool to be used for Eclipse plugin development that focuses on modeling software is IBM RSA. IBM Rational Software Architect is a modeling and development environment that uses the UML for designing architecture for C++ and JEE applications and web services. Rational Software Architect is built in the Eclipse open-source software framework and includes capabilities focused on architectural code analysis, C++, and MDD with the UML for creating applications and web services [IBM, 2015]. Used version is Rational Software Architect 7.5.5.2 that is based on Eclipse 3.4 Ganymede. Since RSA is Eclipse-based, it can take advantage of the market of third-party plugins for Eclipse, as well as plugins specifically for Rational tools.

The tool to be used for Model-to-Model transformation is medini QVT [OMG, 2015], [QVT, 2016]. Medini QVT is a tool set for Model-to-Model transformations [Medini QVT, 2016]. The core engine implements OMG's QVT Relations standard, and is licensed under EPL. Medini QVT also includes tools for convenient development of transformations, such as a graphical debugger and an editor. These components are freely available for non-commercial use only.

Medini QVT has the following features:

- execution of QVT transformations expressed in the textual concrete syntax of the Relations language;
- Eclipse integration;
- editor with code assistant;
- debugger to step through the relations;
- trace management enabling incremental update during re-transformation.

Key concept enabling incremental update as well as the transition from manual modeling to automation through transformations in the absence of traces.

It is a challenge to integrate medini QVT into RSA since simply copying the plugins into the “plugins” folder will not make RSA discover them. The less efficient and safe solution is to use a separate Eclipse 3.6 Helios distribution and install medini QVT in there from an update site. Then in RSA the folder with the Eclipse 3.6 distribution can serve as a local repository of plugins for RSA. By navigating to “Help” menu, and choosing “Software Updates…” option we can switch to “Available Software” tab and click “Add Site…” button. There we can point to an Eclipse distribution with medini QVT installed by clicking “Local…” button. Then in an appeared site option we can chose “QVT Cockpit Feature” and click “Install…”. After that it is a simple matter of clicking “Next” or “Finish”.

The project is an Eclipse plugin project that has several dependencies. Among them are the following required plugins:

Table 3 Description of used Eclipse plugins to plug-in project dependencies

Plugin name	Plugin features
1	2
org.eclipse.ui	Application programming interfaces for interaction with and extension of the Eclipse Platform User Interface.
org.eclipse.core.runtime	Provides support for the runtime platform, core utility methods and the extension registry.
com.ibm.xtools.modeler.ui	UML Modeler primary package. This package exposes the entry point for the UML Modeler API though UMLModeler static class.
de.ikv.medini.qvt.plugin	Contains QVT-R transformation engine.

Table 4. Description of imported packages plug-in project dependencies

Package name	Package features
com.ibm.xtools.modeler.ui	The UML Modeler API consists of a single static utility class, UMLModeler, and of several other classes and interfaces that are accessible from UMLModeler.
com.ibm.xtools.uml.ui.diagram	This package provides the UML modeling API that enables UML diagrams to be created and modified.
com.ibm.xtools.umlnotation	UML Notation meta-model primary package.
org.eclipse.emf.common.util	Provides basic utilities.
org.eclipse.emf.ecore	Provides an API for the Ecore dialect of UML.
org.eclipse.emf.ecore.resource	Provides an API for modeling abstract persistent resources.
org.eclipse.emf.ecore.resource.impl	Provides an implementation of the resource API.
org.eclipse.emf.ecore.xmi.impl	Provides implementation of XMI utility
org.eclipse.gmf.runtime.notation	Provides a standard EMF notational meta model.
org.eclipse.uml2.uml	Provides an API for an EMF-based implementation of the UML 2.5 metamodel for the Eclipse platform

Desciption of Medini QVT transformation script

Let's consider a use case model first. Just as a class model use case model has a root level contained called Package. Naturally it can be mapped into a Package within a class model. Then we might consider an Actor. Actor can be mapped into a Class that implements an Interface. This implementation reflects a Realization relationship element. Then there is a UseCase. Being connected to an Actor through the Association relationship it can represent an Operator owned by an Interface.

The following table summarizes the rules of transformation from use case model to class model:

Table 5. Transformation rules one-to-one from UML Use Case Diagram to class one

Use Case Model	Class Model
Package	Package
Actor	Class, Interface, Realization
UseCase	Operation

The second case of a source model is communication model. Yet again we consider a Package element that is mapped into a similar Package element in a target class model. Then a Lifeline of types Actor, Class, and Component can be mapped into a Class of a class model. Then a Message passed between lifelines can be mapped to an Operation inside of a Class.

The following table summarizes the rules of transformation from communication model to class model:

Table 5. Transformation rules one-to-one from UML Communication Diagram to class one

Communication Model	Class Model
Package	Package
Lifeline (Actor, Class, Component)	Class
Message	Operation

Description of QVT-R Script

Here is a short overview of the integral classes written for a plugin application to support its functionality.

A custom class that represents a transformation engine is called Medini QVTTransformationEngine and it implements custom TransformationEngine interface. It relies on third-party EMFQvtProcessorImpl class to evaluate QVT transformation. An abstract custom class ModelTransformationTemplate conforms to a Template design pattern. It encapsulates a skeleton of an algorithm that describes the model transformation process. The algorithm consists of four steps:

– prepare source model placeholder;
– prepare target model placeholder;
– perform transformation;
– obtain target model.

These four steps are represented by methods and these methods are invoked in a template method named as transformSourceModel of a ModelTransformationTemplate class.

A custom concrete class that extends ModelTransformationTemplate is called Medini QVTRsaDomModelTransformation. The "Medini QVT" part of a class name means that the transformation process is performed by this model transformation tool. The "Rsa" part of the name means that the source model is provided by the Rational Software Architect and a diagram in a target model is created using this tool also. The "Dom" part in the name specifies that the DOM method of XML editing is used.

```
transformation usecase2class (source : uml, target : uml) {
	top relation model2model {
		theName : String;
		checkonly domain source s: uml::Package {
			name = theName
		};
		enforce domain target t: uml::Package {
			name = 'ClassModel_from_' + theName
		};
		where {
			actor2class(s, t);
		}
	}
	relation actor2class {
		modelName : String;
		checkonly domain source sourcePackage: uml::Package {
			packagedElement = actor: uml::Actor {
				name = modelName
			}
```

```
		};
		enforce domain target targetPackage: uml::Package {
			packagedElement = class: uml::Class {
				name = modelName + 'Impl'
			},
			packagedElement = interface: uml::Interface {
				name = modelName
			},
			packagedElement = realization: uml::Realization {
				client = client -> append(class),
				supplier = supplier -> append(interface)
			}
		};
		where {
			usecase2operation(actor, sourcePackage, interface);
		}
	}
	relation usecase2operation {
		usecaseName : String;
		checkonly domain source actor : uml::Actor {
		};
		checkonly domain source s : uml::Package {
			packagedElement = usecase : uml::UseCase {
				name = usecaseName
			},
			packagedElement = association : uml::Association {
				ownedEnd = usecaseEnd : uml::Property {
					name = usecaseName.firstToLower()
				},
				ownedEnd = actorEnd : uml::Property {
					name = actor.name.firstToLower()
				}
			}
		};
		enforce domain target interface : uml::Interface {
			ownedOperation = operation : uml::Operation {
				name = usecaseName
			}
		};
	}
}
```

The following listing describes how to transform a communication model into class model:

```
transformation communication2class (source : uml, target : uml) {
	top relation package2package {
		theName : String;
		checkonly domain source s : uml::Package {
			name = theName,
```

```
                packagedElement = collaboration : uml::Collaboration {
                }
        };
        enforce domain target t : uml::Package {
                name = 'ClassModel_from_' + theName
        };
        where {
                lifeline2class(collaboration, s, t);
        }
}
relation lifeline2class {
        modelName : String;
        checkonly domain source collaboration : uml::Collaboration {
        };
        checkonly domain source sourcePackage : uml::Package {
                packagedElement = lifeline : uml::Classifier {
                        name = modelName
                }
        };
        enforce domain target targetPackage : uml::Package {
                packagedElement = class : uml::Class {
                        name = modelName
                }
        };
        when {
                lifeline.oclIsKindOf(uml::Actor) or lifeline.oclIsKindOf(uml::Class);
        }
        where {
                message2operation(lifeline, collaboration, class);
        }
}
relation message2operation {
        operationName : String;
        checkonly domain source lifeline : uml::Classifier {
        };
        checkonly domain source collaboration : uml::Collaboration {
                ownedBehavior = interaction : uml::Interaction {
                        ownedConnector = link : uml::Connector {
                                end = endPoint : uml::ConnectorEnd {
                                        role = attribute
                                }
                        },
                        message = theMessage : uml::Message {
                                name = operationName,
                                connector = link
                        }
                },
                ownedAttribute = attribute : uml::Property {
                        type = lifeline
```

```
                }
            };
            enforce domain target class : uml::Class {
                ownedOperation = operation : uml::Operation {
                    name = operationName
                }
            };
            when {
                link.allOwnedElements()->at(2) = endPoint;
                not class.allOwnedElements() ->
                select(x | x.oclIsKindOf(uml::Operation)) ->
                collect(y | y.oclAsType(uml::Operation)) ->
                exists(o | o.name = operationName);
            }
        }
}
```

Execution example

Explain how to perform the next points of proposed approach by means of designed script.

Figure1 reflects a dialog window that is part of the plugin and serves as a graphical user interface for the interaction with a user. Such dialog can be created using a standard Eclipse plugin development project.

It is better to execute transformations described by a QVT-R script in an environment that is used for the creation of models. A modeling tool for this purpose is IBM Rational Software Architect. The version used in this paper is IBM RSASE 7.5.5.2. Since Rational Software Architect does not support UML-to-UML transformations there is a need to develop a plugin for the automation of such a task. The plugin has a dependency on another plugin that contains medini QVT transformation engine. So there is a need to integrate the engine into Rational Software Architect.

Since medini QVT 1.7 is based on Eclipse 3.6 Helios and IBM Rational Software Architect 7.5.5.2 is based on Eclipse 3.4 Ganymede the two of them are incompatible. The solution is to install medini QVT into a separate Eclipse 3.6 distribution. Then using this distribution create a feature that would include a plugin containing transformation engine. After that create a local update site and add created feature into it. This update site can be used by Rational Software Architect to install created feature and therefore integrate desired transformation engine for the use by other plugin.

The plugin that transforms UML models with a help of medini QVT transformation engine requires special project structure in order to transform models. Let’s a consider a project folder in RSA environment where the input and output of a transformation are going to be stored. This folder should have three subfolders called “qvt”, “temp” and “traces”. The “qvt” folder is supposed to contain necessary QVT-R scripts. The “temp” folder is supposed to contain temporary file placeholders for source and target models. The “traces” folder will contain the files that are called traces. Traces are generated after the transformation and they store the records of how the transformation took place and which elements participated in a transformation for the creation of target model elements.

The first input field requires a path to a script that describes the rules for the transformation. This must be a QVT-R script. The second field requires a path to an input model to be transformed. The "Browse…" buttons help to find necessary file by opening a file dialog. The "OK" button starts the transformation. "Cancel" button is self-explanatory.

In order to provide an input model into the process of transformation a communication diagram is built based on the plugin's architecture.

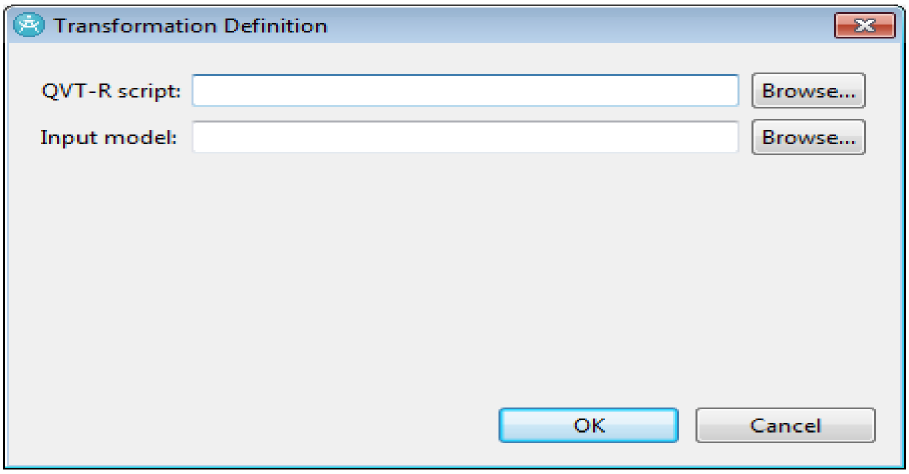


Figure 1. Plug-in window

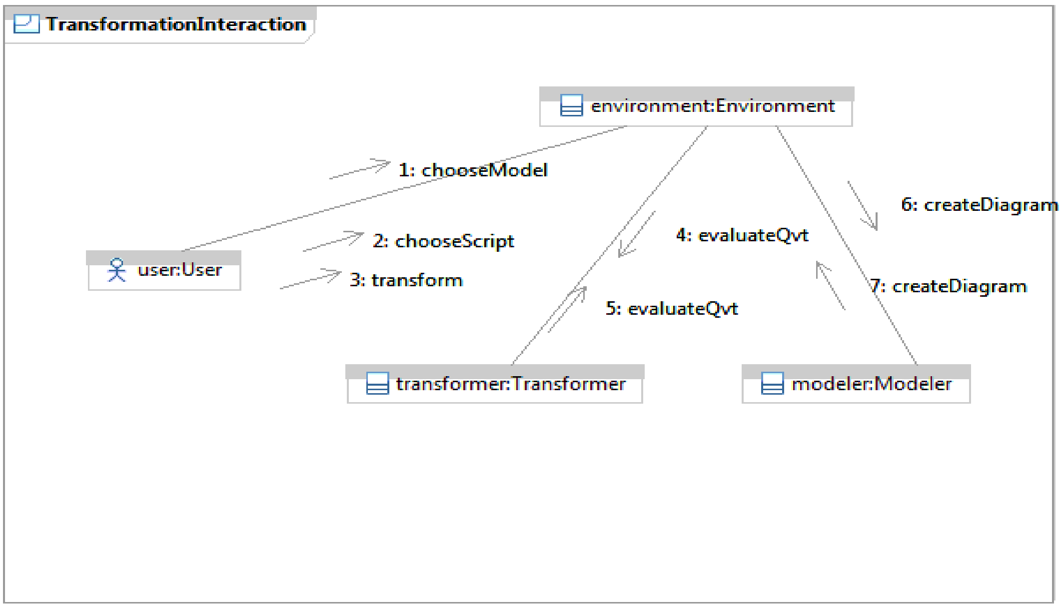


Figure 2. Communication diagram

Communication diagram, represented in Figure 2 is a source model. Particularly there is a "User" actor and three classes "Environment", "Transformer" and "Modeler". Each of them have their own type. The interaction starts with the User lifeline being connected to the Environment lifeline, and in turn Environment is connected to the Transformer and Modeler lifelines. Connections are represented by the message paths through which the messages can be passed back and forth. Specifically, User can send such messages as "chooseModel", "chooseScript" and "transform". Environment can send "evaluateQvt" message to Transformer and "createDiagram" message to Modeler.

The obtained target model represents a class model with generated class diagram.

Fig. 3. Class diagram

In general, there are apparent relationships between lifelines being mapped into classes and messages being mapped into operations.

Conclusion

An approach for Model-to-Model transformation and software tool for class diagram designing are represented in this paper. All variants for one-to-one transformation for UML Use Case, Communication and Sequence diagrams are considered in tables 1 and 2. Such transformations provide a foundation for implementing QVT-R scripts in Medini QVT. The template for QVT-R script for performing one-to-one transformation is proposed in this paper. Using these foundations and QVT-R script templates for one-to-one transformation other transformations can be implemented. An approach for implementation of Model-to-Model transformation when medini QVT plug-ins are built in the IBM Rational Software Architect 7.5.5.2 is represented. Source and target software models are opened in IBM RSA 7.5.5.2.

Designed software tool allows automating the task of creating a class diagrams from the communication diagrams. Flexibility of designed tool allows changing rules of script and scripts themselves (fig. 1) for interoperating with software models of different types. It allows to high skill specialist to formulate specific transformation rules that reflect peculiarities of problem domain and experience of a specialist.

Other advantage of designed plug-in that source and target software models are defined as external parameters for plug-in execution.

Implementing this plugin-in into software development life cycle increases the effectively of all software development where Model-to-Model transformation is performed.

Further Research

Design an approach and software tool, supporting Model-to-Model transformation, integrating all types of transformation rules, namely many-to-many, one-to-many, and many-to-one. It allows to realize ideas, presented in the papers [Chebanyuk, 2014a] and [Chebanyuk, 2014b]. Integrate transformation from Communication diagram to class one and verification of obtained class diagram by means of analytics, represented in [Chebanyuk and Markov, 2016b].

Acknowledgements

The paper is published with partial support by the ITHEA ISS (www.ithea.org) and the ADUIS (www.aduis.com.ua)

Bibliography

[Balasubramanian et al, 2007] Balasubramanian, D., Narayanan, A., van Buskirk, C., Karsai, G. The graph rewriting and transformation language: GReAT. *Electronic Communications of the EASST*, *1, 2007*.

[Chebanyuk and Markov, 2016a] Elena Chebanyuk, Krassimir Markov. Model of problem domain “Model-driven architecture formal methods and approaches”. *International Journal “Information Content and Processing”, Vol. 2, No.3, ITHEA 2016, p.202-222.*

[Chebanyuk and Markov, 2016b] Elena Chebanyuk, Krassimir Markov (2016). An Approach to Class Diagrams Verification According to SOLID Design Principles.In Proceedings of the 4th International Conference on Model-Driven Engineering and Software Development - Volume 1: MODELSWARD, ISBN 978-989-758-168-7, pages 435-441. DOI: 10.5220/0005830104350441

[Chebanyuk, 2014a] Chebanyuk E. Method of behavioral software models synchronization. *International Journal “Informational models and analysis” – 2014, №2, pp. 147-163. http://www.foibg.com/ijima/vol03/ijima03-02-p05.pdf*

[Chebanyuk, 2014b] Chebanyuk Elena. An approach to class diagram design. 2nd International Conference on Model-Driven Engineering and Software Development (MODELSWARD), Lisbon, Portugal, 2014, pp. 448-453. Access mode:

[Chebanyuk, 2015] Chebanyuk Elena. An approach to behavioral software models analytical representation. *International journal Informational models and analysis, 2015, Vol.3, №1, pp. 61-79 http://www.foibg.com/ijima/vol03/ijima03-02-p05.pdf*

[Chomsky, 1957] Chomsky, Noam. Syntactic Structures. Mouton publishers, Eilenberg: Mac Lane The, Hague*, 1945 - 1957. ISBN 90 279 3385 5. 1957 p.107. http://ewan.website/egg-course-1/readings/syntactic_structures.pdf*

[Cicchetti et al, 2010] Cicchetti, A., Di Ruscio, D., Eramo, R., & Pierantonio, A. (2010, October). JTL: a bidirectional and change propagating transformation language. In International Conference on Software Language Engineering (pp. 183-202). Springer Berlin Heidelberg.

[Eclipse, 2016] ATL Transformation Language http://www.eclipse.org/atl/

[Errata et al, 2015] Ferhat Errata, Moharram Challenger, Geylani Kardas, Moharram Challenger, Geylani Kardas, Review of Model-to-Model Transformation Approaches and Technologies, 2015 Mode of access: https://itea3.org/project/workpackage/document/download/2305/13028-ModelWriter-WP-3-D311ReviewofModel-to-ModelTransformationApproachesandTechnologies

[Favre and Duarte, 2016] Favre, Liliana and Daniel Duarte. 2016. Formal MOF Metamodeling and Tool Support. *In: MODELSWARD 2016, Proceedings of the 4th International Conference on Model-Driven Engineering and Software Development. Edited by S. Hammoudi, L.F. Pires, B. Selic and P. Desfray. SCITEPRESS – Science and Technology Publications, Lda. Portugal, 2016. ISBN: 978-989-758-168-7. pp. 99-110. DOI:10.5220/0005689200990110, http://www.scitepress.org/DigitalLibrary/ProceedingsDetails.aspx?ID=j1i7qrX33Ns=&t=1*

[Greiner et al, 2016] Greiner Sandra, Thomas Buchmann, Bernhard Westfechtel. 2016. Bidirectional Transformations with QVT-R: A Case Study in Round-trip Engineering UML Class Models and Java Source Code. *In: MODELSWARD 2016, Proceedings of the 4th International Conference on Model-Driven Engineering and Software Development. Edited by S. Hammoudi, L.F. Pires, B. Selic and P. Desfray. SCITEPRESS – Science and Technology Publications, Lda. Portugal, 2016. ISBN: 978-989-758-168-7. pp. 15-27. DOI:10.5220/0005644700150027 http://www.scitepress.org/DigitalLibrary/PublicationsDetail.aspx?ID=efZXth7Zbbg=&t=1*

[IBM, 2015] IBM Rational Software Architect (2015) – Mode of access: https://en.wikipedia.org/wiki/Rational_Software_Architect

[Kuster, 2011] Jochen Küster. Model-Driven Software Engineering Model Transformations II IBM Research (2011) – Zurich. access mode http://researcher.ibm.com/researcher/files/zurich-jku/mdse-07.pdf

[Lara and Vangheluwe, 2002] Lara J., Vangheluwe H. (2002) AToM3: A Tool for Multi-formalism and Meta-modelling. In: Kutsche RD., Weber H. (eds) Fundamental Approaches to Software Engineering. FASE 2002. Lecture Notes in Computer Science, vol 2306. Springer, Berlin, Heidelberg

[Macedo and Cunha, 2013] Macedo, N., Cunha, A. Implementing QVT-R bidirectional model transformations using Alloy. In *International Conference on Fundamental Approaches to Software Engineering* 2013 (pp. 297-311). Springer Berlin Heidelberg.

[Massito et al, 2009] Tisi, M., Jouault, F., Fraternali, P., Ceri, S., & Bézivin, J. (2009, June). On the use of higher-order model transformations. In European Conference on Model Driven Architecture-Foundations and Applications (pp. 18-33). Springer Berlin Heidelberg. 2009

[Medini QVT, 2016] access mode http://projects.ikv.de/qvt

[Natschlager et al, 2016] Natschlager, Christine, Verena Geist, Christa Illibauer and Robert Hutter. 2016. Modelling Business Process Variants using Graph Transformation Rules *In: MODELSWARD 2016, Proceedings of the 4th International Conference on Model-Driven Engineering and Software Development. Edited by S. Hammoudi, L.F. Pires, B. Selic and P. Desfray. SCITEPRESS – Science and Technology Publications, Lda. Portugal, 2016. ISBN: 978-989-758-168-7. pp. 65-74. DOI:10.5220/0005686900870098, http://www.scitepress.org/DigitalLibrary/PublicationsDetail.aspx?ID=IzjjeczBZuA=&t=1*

[OMG, 2015] XML Metadata Interchange http://www.omg.org/spec/XMI/2.5.1/

[QVT, 2016] Meta Object Facility Query/View/Transformation, v1.3

[Rabbi et al, 2016] Rabbi, Fazle, Yngve Lamo, Ingrid Chieh Yu, Lars Michael Kristensen. 2016. WebDPF: A Web-based Metamodelling and Model Transformation Environment. *In: MODELSWARD 2016, Proceedings of the 4th International Conference on Model-Driven Engineering and Software Development. Edited by S. Hammoudi, L.F. Pires, B. Selic and P. Desfray. SCITEPRESS – Science and Technology Publications, Lda. Portugal, 2016. ISBN: 978-989-758-168-7. pp. 87-98. DOI:10.5220/0005686900870098, http://www.scitepress.org/DigitalLibrary/PublicationsDetail.aspx?ID=IzjjeczBZuA=&t=1*

[Rebull et al, 2007] Rebull, E. V. S., Avila-Garcıa, O., Garcıa, J. L. R., Garcıa, A. E. (2007). Applying a modeldriven approach to model transformation development.

[Seifermann and Groenda, 2016] Seifermann, Stephan and Henning Groenda. Survey on Textual Notations for the Unified Modeling Language *In: MODELSWARD 2016, Proceedings of the 4th International Conference on Model-Driven Engineering and Software Development. Edited by S. Hammoudi, L.F. Pires, B. Selic and P. Desfray. SCITEPRESS – Science and Technology Publications, Lda. Portugal, 2016. ISBN: 978-989-758-168-7. pp. 28-39. DOI:10.5220/0005686900870098,*

[Wiemann, 1995] Wiemann, H. (1995). Theory of Graph Transformations access mode: http://www.informatik.uni-bremen.de/st/lehre/Arte-fakt/Seminar/Ausarbeitungen/04_Theorie_Graphtransformationen.pdf

[Wu, 2016] Wu, Hao Generating Metamodel Instances Satisfying Coverage Criteria via SMT Solving *In: MODELSWARD 2016, Proceedings of the 4th International Conference on Model-Driven Engineering and Software Development. Edited by S. Hammoudi, L.F. Pires, B. Selic and P. Desfray. SCITEPRESS – Science and Technology Publications, Lda. Portugal, 2016. ISBN: 978-989-758-168-7. pp. 40-51. DOI:10.5220/0005686900870098,*

Authors' Information

Elena Chebanyuk *– Assoc. Prof. of Software Engineering Department, National Aviation University, Kyiv, Ukraine,*

e-mail: *chebanyuk.elena@ithea.org*

Major Fields of Scientific Research*: Model-Driven Architecture, Model-Driven Development, Software architecture, Mobile development, Software development*

Kyryl Shestakov *– Software Engineering Department, National Aviation University, Kyiv, Ukraine,*

e-mail: *KyrylO.Shestakov@livenau.net*

Major Fields of Scientific Research*: Model-Driven Architecture, Model-Driven Development, Software architecture, Mobile development, Software development*

INCORRECT MOVES AND TESTABLE STATES

Dimiter Dobrev

***Abstract**: How do we describe the invisible? Let's take a sequence: input, output, input, output ... Behind this sequence stands a world and the sequence of its internal states. We do not see the internal state of the world, but only a part of it. To describe that part which is invisible, we will use the concept of "incorrect move" and its generalization "testable state". Thus, we will reduce the problem of partial observability to the problem of full observability.*

***Keywords**: Artificial Intelligence, Machine Learning, Reinforcement Learning.*

***ITHEA Keywords**: I. Computing Methodologies: I.2 Artificial Intelligence: I.2.6 Learning*

Introduction

Our first task in the field of Reinforcement learning is to describe the current state of the world (or the current position of the game). When we see everything (full observability) this question does not arise because in this case the input fully describes the state of the world. More interesting is the case when we see only a part of the world (partial observability). Then we will add more coordinates to the input vector, and thus we get a new vector which already fully describes the state of the world.

To add these new coordinates we will first introduce the concept of "incorrect move". The idea that not all moves are correct is not new. Other authors suggest, for example, that you cannot spend more money than you have. That is, they assume that the output is limited by a parameter that changes over time. What's new in this article is that we will use this parameter to give further description of the state of the world.

If at a specific moment we know what we see and what moves are correct at that specific moment, we know a lot about the current state of the world, yet we do not know everything. When we generalize the concept of "incorrect move", we will derive the new concept of "testable state". If we add to the input vector the values of all testable states, we will get an infinite-dimensional vector that fully describes the state of the world. That is, for every two distinct states there will be a testable state whose value in these two states is different.

Incorrect moves and testable states are something that actually exists just like the input we get on each step. However, unlike the input, the value of testable states is not ready to derive. For example, is the door locked? To check this we need to push the handle and see whether the door will open, but we can do it only if we stand by the door. In other words, the check requires extra effort and it is not always possible (there are moments in which we can check and moments in which we can't). The locked door can be considered as an example of both an incorrect move and of a testable state.

To describe the incorrect moves and testable states we will search for a theory that gives us this description. Of course, we may have many theories for a given testable state all of which will compete with each other over time.

What will the theory constitute? Statistics shows us that in specific situations a given testable state is true. Specific situations means a situation in which a conjunction is true. This conjunction may be associated only with the past, but may also be associated with the past and the future. For example, let's say we've checked and we've seen that a door is locked, but is it the door that we are interested in? We may decide that it is precisely this door on the basis of what we have seen before checking, or maybe after checking, a posteriori, we've seen something which tells us that it is exactly this door.

Another generalization of "specific situations" will be to allow dependencies with memory (except dependencies without memory). A dependency without memory is the situation in which specific events occur in the course of several consecutive steps (i.e. this is an ordinary conjunction). A dependency with memory is the situation in which specific events occur at specific moments (not consecutive), and in the periods between those moments specific events do not occur.

The theory may be a set of such implications and this set can be generalized as a finite state machine. Let's take an example where we are moving in a castle in which some of the doors are locked. We can have a rule of the following type: "*If you see a white sofa and you turn right, then the door will be locked.*" If we represent the map of the castle as a finite state machine, we will see that if after the white sofa we turn right, we are at the door *X* and that this door is locked. If we know the finite state machine, we can easily derive the corresponding rules. Unfortunately, we need the opposite – to derive a finite state machine from the rules, and that's a much more difficult task.

Let us now imagine a castle the doors of which are not permanently locked or unlocked but change following specific rules. Then our theory will suggest some sustainability of testable states. For example, if a door has been locked at a specific moment and shortly thereafter we check again, we assume that it will be locked again, especially if during this time no specific events have occurred (for example, to hear a click of door locks).

The next upgrade of the theory would be to assume that there is some kind of creature inside the castle, which unlocks and locks the doors in its sole discretion. Then we can predict whether a door is locked or unlocked predicting the behavior of that creature.

Once we've created one or several theories that predict a testable state, we will use these theories and gather statistics that will help us predict the future and to create new theories of other testable states.

For example, in our case, if we've noticed that behind the locked door there is a princess, and a tiger behind the unlocked door, then based on the theory that tells us which door is locked, we can make a theory that tells us where the princesses are.

Definitions

Let's take a sequence of *output, input, output, input, … ,* and the goal is to understand this sequence.

Of course, this sequence is not accidental. We can assume that we are playing a game and that's the sequence:

move, observation, move, observation ...

And our goal is to understand the rules of the game and what is the current position on the game board.

We might assume that we are in a world and then the sequence would be:

action, view, action, view ...

In this case, our goal is to understand the rules of this world and what is the current state of the world.

The description of the world is given by the functions *World* and *View*, and the following applies:

$s_{i+1}=World(s_i , a_{i+1})$

$v_i =View(s_i)$

Here, actions and observations (a_i and v_i) are vectors from scalars with dimensions n and m, respectively.

Our goal is to present the internal state (s_i) also as a vector of scalars, in which the first m coordinates will be exactly v_i. In other words, the function *View* will be the projective function that returns the first m coordinates of s_i.

We will call "states" to all coordinates of s_i. We will call the first m ones "visible states". Other coordinates will be called "testable states".

The coordinates of the input and output will be called "signals". These are functions of time that return a scalar. To the input and output signals we will add other signals as well, like the testable states; more precisely – the value of the testable state according to the relevant theory, because we will not know the exact value of these states, and will approximate them with some theories. For each finite state machine we will add a signal whose value will be equal to the state in which the finite state machine is situated at the moment t. Of course, if the machine is nondeterministic, the value of this signal may not be exact but approximate.

We will call the Boolean signals "events ". When the Boolean signal is truth, we will say that the event is happening.

Example

To make things clear, let’s take an example. Let’s imagine we are playing chess with an imaginary opponent without seeing the board (we are playing blind). We will not specify whether the imaginary opponent is human or a computer program.

Our move will be the following 4-dimensional vector:

X_1, Y_1, X_2, Y_2

The meaning is that we are moving the piece from the (X_1, Y_1) coordinates to the (X_2, Y_2) coordinates.

What we will see after each move is a 5-dimensional vector:

A_1, B_1, A_2, B_2, R

The first four coordinates of the input show us the counter-move of the imaginary opponent, and *R* shows us our immediate reward.

All scalars have values from 1 to 8, except *R*, whose value is in the *{-1, 0, 1, nothing}* set. The meaning of these values is as follows: *{loss, draw, win, the game is not over}*.

Incorrect move

Shall we allow the existence of incorrect moves?

Our first suggestion is to choose a world in which all moves are correct. In the example we took, we cannot move the bishop as a knight, so it is natural to assume that some of our moves are incorrect or impossible.

Our second suggestion is to have incorrect moves and when we play such a move to assume that the world penalizes us with a loss. So we will very quickly learn to avoid incorrect moves, but here we are denied the opportunity to study the world by checking which move is correct (like touching the walls in the darkness).

Our third suggestion is: When an incorrect move is made the state of the world to remain the same, and instead of “loss” the immediate reward to be "incorrect move" and all other coordinates at the input to return "nothing". This option is better, but thus we would unnecessarily prolong the history. (We will call "history" the following sequence: $a_1, v_1, \ldots, a_{t-1}, v_{t-1}$, where *t* is the current time). Given that the state of the world remains the same, there is no need to increase the count of the steps.

The fourth suggestion is: When you play an incorrect move, you to be informed that the move is incorrect but without prolongation of the history. The new history will look like this: $u_1, a_1, v_1, \ldots, u_t, a_t, v_t$. Here u_i is the set of incorrect moves at the *i*-th step which we have tried and we know for sure that they are incorrect. Thus the history is not prolonged, yet the information that certain moves are incorrect is recorded in the history. Here we assume that the order in which we've tried the incorrect moves does not matter.

The fifth option, which we will discuss in this article, will be even more liberal. In the previous suggestion we have the opportunity to try whether a move is incorrect, but if it proves correct, we have already made this move. Now we will assume that we can try different moves as many as we want and we can get information whether it is correct or incorrect for each one of them. After that, we make a move, for which we already know that it is correct. We know because we've tried it. Here, the history is the same as with suggestion No. 5, except that u_i is a tuple of two sets, the first of which is from the tried incorrect moves, the second – from the tried correct moves.

There is a sixth option and it is for every step to get full information about which moves are correct and which are not. In other words, we get u_i with all the moves in it like they have been tested. However, we do not like this option because u_i may be too large, i.e. it may contain too much information. Moreover, we assume that after some training we will have a fairly accurate idea about which move is correct and which is incorrect and will not have to make many tests in order to understand this.

Conclusion

In the field of AI our first task is to say what we are looking for, and the second task is to find the thing we are looking for. This article is entirely devoted to the second task.

Articles [Dobrev, 2000, 2005, 2013] and [Legg, 2005] which are devoted to the first task, tell us that the thing we are looking for is an intelligent program that proves its intelligence by demonstrating it can understand an unknown world and cope in it to a sufficient degree.

The key element in this article is the understanding of the world itself, and more precisely – the understanding of the state of the world. We argue that if we understand the state of the world, we understand the world itself. Formally speaking, to understand the state of the world is like reducing the problem for Reinforcement learning with partial observability to the problem for Reinforcement learning with full observability. Not accidentally, almost all articles on Reinforcement learning deal with the case of full observability. This is because it is the easiest case. The only problem in this case is to overcome the combinatorial explosion. Of course, combinatorial explosion itself is a big enough problem because we get algorithms that would work if you have an endlessly fast computer and indefinitely long time for training (long in terms of the number of steps). Such algorithms are completely useless in practice and their value is only theoretical.

In this article we presented the state of the world as infinite-dimensional vector of the values of all testable states. This seems enough to understand the state of the world, but we need a finite description and we would like this description to be as simple as possible. For this purpose, we've made three additional steps. A model of the world was introduced as a finite state machine. Each such machine describes an infinite number of testable states and this is a very simple description which is easy to operate.

The second step was the assumption that testable states are inert and change only if specific events occur. That is, if between two checks, none of these events has occurred, we can assume that the value of the testable state has not changed.

The third step was the introduction of agents which under certain conditions may alter the values of testable states. This step is particularly important because the agent hides in itself much of the complexity of the world and thus the world becomes much simpler and more understandable. We will not describe the agent as a system of formal rules. Instead, we will approximate it with assumptions such as that it is our ally and tries to help us or that it is an opponent and tries to counteract.

Without these three steps the understanding of a more complex world would be completely impossible. Formally speaking, testable states only would be sufficient to understand the state of the world. Even testable states which prerequisites are simple conjunction dependent only on the last few steps of the past are sufficient. However, without the additional generalizations made, we would face an enormous combinatorial explosion.

Bibliography

[Dobrev, 2000] Dobrev D. AI – What is this. PC Magazine – Bulgaria, November'2000, pp.12-13. http://www.dobrev.com/AI/definition.html

[Dobrev, 2005] Dobrev D. Formal Definition of Artificial Intelligence, International Journal "Information Theories & Applications", vol.12, Number 3, 2005, pp.277-285. http://www.dobrev.com/AI/

[Dobrev, 2013] Dobrev D. Comparison between the two definitions of AI. arXiv:1302.0216, January, 2013. http://www.dobrev.com/AI/

[Legg, 2005] S. Legg and M. Hutter. A Universal Measure of Intelligence for Artificial Agents, Proc. 21st International Joint Conf. on Artificial Intelligence (IJCAI-2005), pages 1509–1510, Edinburgh, 2005.

Authors' Information

Dimiter Dobrev *– Institute of Mathematics and Informatics, Bulgarian Academy of Sciences. Acad. Georgi Bonchev Str., Block 8, 1113 Sofia, Bulgaria; e-mail: d@dobrev.com*

Major Fields of Scientific Research: Artificial Intelligence, Logic Programming

FUTURE OF ARTIFICIAL INTELLIGENCE: TREATS AND POSSIBILITY.

Volodymyr Hrytsyk

Abstract: *In this article, author wants to focus on problem that are important and urgent in the sense of their solutions are needed before to the artificial intelligence explosion. Some technical problems of human – robots cooperation in the field of artificial Intelligence are searched. The research needed to reduce the risks of the machine intelligence revolution; author will propose his vision of threats and possibility. Author tried to show some examples of Independence from humans of artificial Intelligence systems and humans needs of it. Author want to show: people strongly needs to find their place in co-exist with other form of life that will be much smarter than people and will be more dangerous, more usefully for humans according to their smart.*

Keywords: *Artificial Intelligence, future.*

ITHEA Keywords: *computer science, informatics, cybernetics, control theory, information technologies, control systems, information systems, industrial*

Introduction

Artificial Intelligence problems that were unrelieved to solve before, thus are solved now. Knowledge of deep learning algorithms did possible to create voice recognition systems. Self-driving cars – it is reality. Computer vision and computer sensibility start working. Machine analyzing, planning, creating, intelligence decision making - all those show us that theoretical and applied methods and realizations are now, when we can prognosis: people will lose control for decision making at least 99% of it. Intelligence and not machines will take the control from humans. Some people out of IT would say: "this prognosis is too fantastic, we have much time to such futuristic future". But it would be mistake – I think it can be single possible of future. Impact to this study was combination of next factors:

- At first – it the rapid development of new applied methods of Computer thinking and perception;
- At second – it fourth Industry revolution (Industry 4.0): interpenetration IT and production technologies (automation, data exchange in manufacturing technologies …) is on the level were people participation is more rapidly weakening economic component then technological component. We can add to part of this level other areas: services, finance, insurance… We can see up to 2030 in the World human – insurance, finance it will be history. We will buy most goods and services without humans help.

Task and challenges

We study all factors in the paper and propose possible solutions. Aim of the paper is searching and plan possible ways to co-exist of humans and artificial intelligence (AI) systems. Possibility of creating of software and hardware much more productivity and creativity then humans (in next 20 years) makes important this work. So we need to try looking for solution problem of co-existing with other form of life will be on the top of evolution chains.

Study

Construct a sequence of production technologies. The first industrial revolution had had led to the transition from manual to mechanized production by using steam engine. The second industrial revolution had led to the transition to mass production by the use of an electric motor and conveyor. Third Industrial Revolution led to the transition to automated production by the use of computer and information technology [1,2,4,6,7]..

For the first time the term «Industrie 4.0» became known in April 2011, when the Hanover Fair group «Industrie 4.0» took the initiative to enhance the competitiveness of the German economy [3, 6].

Definition:

The fourth industrial revolution [1] [2] – it is concept of development and merging automated production, exchange and production technologies into a single self-regulating system with a minimal or no human intervention in the production process [5].

The concept is "a collective term for technologies and concepts of organization added value chain" by using cyber-physical systems, Internet of Things, Internet services, smart factories [6]. Therefore, we speak about fusion of technologies between physical, digital and biological areas [7]. Industry 4.0 concept assumes smart machines will connect between themselves, will correct themselves, and will study new behavior models without people. It will lead to production and services with less mistakes and better adaptation to new needs of users without people. As example – Tesla car can adapt to user by monitoring his behavior and priority.

Today, when Industry 4.0 – it is reality, we can see nearest future: in a chain of raw materials - production - sales - delivery - the customer, the person may become unnecessary. Today, we see a break down the last bastions of humanity competitive advantage - creation, development, design, analysis, planning and so on. Now it looks just as bright pictures of break news: computer intelligence won the champion Chinas strategy game [8], or self-driving cars studies user behaviors and correction and planning or robots – service, sales and projecting without people. All these things are out of human competition, at least, because countries – planetary economy engines, smart robotics has pushed people with manufacturing and continues to displace other industries.

This trend of evolutionary change military does not remain aloof: self-guided weapon that is designed to respond quickly to new situations already replacing people with melee [Roe, laser system to intercept missiles, counter battery etc]. It is not just about air and maritime space, on land terrain examples are military orders [9]. The sooner such systems become, the higher the decision-making chain they move. And - as if on one side of the decision taken, then on the other side, they too must take fast and error free decision. Otherwise, in a split second that the other party will not be easy. With higher speed and the military have no time for making decisions - or decisions are made by a computer or a you are without future. Some example is situation where flying time nuclear missile is less then decision-making chain of other side. The same are correct to missiles and aircraft (even now there will be swarms of bots) - or computer decides, or ... actually it has changed its views on warfare - is developing a strategy of win nuclear war without nuclear weapons. From one side World without nuclear weapons is good idea but price – it is decision making without human.

Alternative Energy 40% leads us to predict the full independence of these systems of people and their opinions (in case of wrong self-correction algorithm) about the need for such systems. Needless recoverability of such systems with modern production systems, navigation and shipping is not a problem. Ahead of them (the independent agents) still expects 3D printing. But returning to the military, the need swarms printing directly on the plane is not entirely fantastic illusion. We can promise not to create autonomous robots for military use (as did Clearpath Robotics), but what will be if it be released by opponents?

In the nearest future, when people will lose jobs in manufacturing, society can provide absolute income [11]. However, in the medium term, if you just pay - it will lead to degradation. And if you put a condition on lifelong learning (which is a good idea), the rule under which the computer will determine the integrity and taught at all - because there is no incentive - computers do everything better, even to plan and implement training can be better. Already, mobile applications can better teach grammar and words than human in grade school can. In the future, schools will be required only for acquiring social skills (author believes that this will be their main objective), as better knowledge will get through IT.

Perhaps the development of social relationships in schools will only strong asset of humanity. Although, it can be modeled. On June 1 on Code Conference Musk said that humanity is already in the matrix: "We have to hope that we all live in the simulator ... or we create simulations that are indistinguishable from reality, or civilization will cease to exist" - then said the engineer. But, how predict the behavior? Complete computer simulation of the whole society is a matrix. The fact that humanity is living in a computer simulation of a probability of 20-50 percent said Bank of America Merrill Lynch to its customers, writes Business Insider [12].

3.1. Robots are coming for our jobs

The Investment bank “Morgan Stanley” predicted that after 8 years the AI systems will manage the assets of $6.5 trillion at the market [13].

Today, there are some still running deterrent social interactions when making decisions. However, the next generation that gets used from school to interact with the work, as is the case with smart phones, this constraint will be gone. IT will develop even faster. Robotic consultants is a program that analyzes the state of a particular sector (finance, transport) and picks the best options for the lowest fee. For example automation in finance has led to the fact that the New York Stock Exchange since 2000, the number of traders decreased from 5500 to 400 people, and the world's largest banks reduce the number of jobs financial managers.

3.2. Robots take manager positions

According to [14] AI effectively displaces workers and personnel decisions. In particular, the use of AI in recruitment, performance and control the release of employees is more effective than HR – human.

3.2.1. Recruitment

To search for new recruits, staff people on average read 300 resumes a day, looking for an option. But, for what? Modern search algorithm sufficient to list the requirements for the employee and the program does the choice of 90%. For example, the algorithm Resume Matcher for teaching reading articles on Wikipedia, which describes a particular profession, duties that must fulfill a particular employee, then the candidates are assigned the label "accept", "reject», «shirt-list».

3.2.2. Evaluation of the effectiveness of the employee

When AI is checking the efficiency of workers it is more effective also. For some employers is important to know what workers do on the job and how to maintain the performance of their work during paid hours. For example, the program can, logs keystrokes, mouse movements, opening certain web-pages at regular intervals photographs status screen. Additionally, the program collects metadata such as date and time, send messages and more. AI analyzes the data and determines the overall efficiency of the office. AI reported about time of best office running efficiently. AI reports on the reasons for the fall of efficiency of the office and warn of danger, for example, someone copied the customer base - wants to leave the company, someone spends a lot of time on porn sites, and someone works hard and can reward.

AI tracks the movement of people in the office, office can be divided into zones and restrict the movement of certain employees. The system can determine exactly how much time the employee spent for jobs, and how many in the hallway / bathroom, restaurant.

Now, AI knows what workers do on computers or office. But, AI goes further - it determines the emotional state employees, today the program analyzing words and phrases from correspondence officer can determine the ratio of staff to work.

Monitoring "loyalty / satisfaction" is always connected HR-specialists only exceptional cases. Program reacts to changes in behavior and tone of behavior. Employee was friendly previously, if suddenly his tone changed, it is signal to the leadership - to solve the problem at an early stage.

3.2.3. Dismissal. When a valuable employee begins to think about the dismissal

Today AI uses behavioral models to study data from thousands of employees to predict when an employee wants to leave the company. The creators of such systems are looking for common patterns of human behavior based on large samples of data that have got by decades. Programs remembers data about employee have been taken, released, or went to work independently. The program calculates the probability based on the signs of employee leaving work. Affect data such as salary, availability of draft, position and so on. AI may propose action for the conservation of frames, for example, to increase wages, or move to another position, to pay a premium.

4. **Agriculture**

If someone thinks that agriculture can be a shelter from AI systems, it is a mistake – the future of agriculture – a complete automation, something like a data center with intelligent sensors that analyze soil (baking mix), the state of plant level lighting, irrigation, fertilizers, food, the air. A sun altogether may be superfluous – lighting can be artificial (Figure 1). This is especially noticeable after the development of the technology of artificial photosynthesis. Plants (grossed by this technology) can absorb carbon dioxide is several times faster [15, 16].

So types farms will be much more effective (Figure 2) for existing today, and people will be completely excluded from chain of the growing – delivery.

When in introduction was wrote "new approaches applied to the perception of reality by computer systems" Author meant Internet of Things. Internet of Things it is the concept of space in which all of the analog and digital worlds can be combined. This is not just a variety of different devices and sensors, combined together wired and wireless communication channels and connected to the Internet that exchange large amounts of data. This is closer integration of real and virtual worlds in which communication is made between people and devices. Internet of things attacked the second driving human trait - the ability to work with other types of environment - bringing this feature to the absolute.

Figure 1. Useful elements horizontally-oriented intelligent farm

Figure 2. Horizontally oriented farm that does not need the sun.

5. AI system and the stress

Recent studies have shown that in order to avoid defeat AI changes aggressive behavior. After AI from DeepMind won puzzle Go game. Google has conducted a study and adapt to conditions which have shown that stressful situations AI is aggressive.

6. Next step: AI no longer needs the support of people for self

AI Deep Mind, no longer needs the support of people for further education. This situation has arisen as a result of the introduction of this system AI system Differential Neural Computer (DNC), which combines large storage of information mechanisms, logical skills AI ANN ability to quickly search the repositories needed fragments.

Now people have artificial intelligence system that no longer requires people to move on to learn ways to solve the problem by themselves and created.

7. Ethics of AI

Asilomar AI Principles were developed in conjunction with the 2017 Asilomar conference [17]:

These principles have three parts: research issues, ethics and values, long-term issues. Each part describes something useful for society.

8. Possible solutions

At Superintelligence book [18] author show us possibility of future control on AI: boxing methods, Incentive methods, stunting, tripwires, motivation selection, direct specification, domesticity, indirect normativity, augmentation.

9. Unexpected situations

Consider this problem with a simple example. Research and analysis of the first decade of Wikipedia showed that a large number of automated software bots (algorithms editing articles), working at the AI - were involved in many disputes concerning the specific editing. Each of the algorithms tried to leave the right to the last editing them.

Shortly, what we speak about? Editorial boats other than editorial amendments, repress vandalism, blocking unscrupulous users create links, check spelling automatically bring new content to the site, etc.

These bots are created to help people, but we give an example, boats war, which took place in 2009-2010. In studies of Dr Taha Yasseri (Oxford University) two boats «Xqbot» and «Darknessbot», which eventually had written over 3,600 articles? Xqbot mentioned during the war in 2000 rejected amendments that introduced Darknessbot. Darknessbot banned in 1700 amendments, which did Xqbot.

Another new about bots: bots behavior dependent on the language and culture. For example, during the same period, the German version of Wikipedia was mentioned least between bots conflicts - about 24 cases per bot on average; in Portuguese version - 185 per bot conflicts; English version - are fighting the average rate - 105 per conflicts bot.

Of course, fighting bots on Wikipedia – not a fatal blow to humanity (we do not generally observed, but note that the editorial bots no effect on any of the critical infrastructure). But, it clearly shows realistic scenario where people (humanity) can get "on hand" in internecine disassembly robots.

10. Conclusion

Artificial Intelligence studies – and studies have shown to an end – he used all available means without emotion. And so, we have not given the kind and type of AI that will dominate in the future (Speed Intelligence, quality intelligence, collective intelligence, brain-computer intelligence, network and organization intelligence or other) we have to adapt and co-exist. The first step is to develop the basic principles of co-existence in the first place for us. Of course, we need some AI ethic:

1. Conditions "do no harm" will identify problems at early stages;
2. Survival of cybernetic system will have better chances if there is diversity: IT, biological systems, other combinations; Success example is the success story of humanity - those that combine and integrate / assimilate are more successful than those that closed
3. Asilomar AI Principles.

According to the author, at this stage, effective response to the explosion of AI may be a combination of unconcentration (AI should not have a single center) and interface (person must be able to interact with arbitrary (s) device (s). Perhaps, as suggested by Elon Musk it will be a man-machine symbiosis. Or maybe it will be virtualization (a kind of digital immortality) when we create digital copies and load them into the machine.

In any case we need to searching and projecting cyber-biological fusion interface.

Bibliography

1. Klaus Schwab. The Fourth Industrial Revolution: what it means, how to respond ((англ.)). World Economic Forum. E-ресурс: https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond;
2. Клаус Шваб. Четверта промислова революція: як до неї готуватися (переклад). Національний університет біоресурсів і природокористування України. E-ресурс: http://nubip.edu.ua/node/23076;
3. H. Kagermann, W.-D. Lukas, W. Wahlster. Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution ((нім.)). VDI nachrichten. E-ресурс: http://www.vdi-

nachrichten.com/Technik-Gesellschaft/Industrie-40-Mit-Internet-Dinge-Weg-4-industriellen-Revolution;

4. Фабіан Шмідт, Захар Бутирський. Ганноверський ярмарок: інтернет зливається з заводом. Deutsche Welle. E-source: http://www.dw.com/uk/ганноверський-ярмарок-інтернет-зливається-з-заводом/a-16728837;
5. Олександр Крамар. Погляд із периферії. E-source: http://tyzhden.ua/Economics/162508;
6. Mario Hermann, Tobias Pentek, Boris Otto. Design Principles for Industrie 4.0 Scenarios: A Literature Review. E-source: http://www.snom.mb.tu-dortmund.de/cms/de/forschung/Arbeitsberichte/Design-Principles-for-Industrie-4_0-Scenarios.pdf;
7. Klaus Schwab. The Fourth Industrial Revolution: what it means, how to respond. E-source: https://www.foreignaffairs.com/articles/2015-12-12/fourth-industrial-revolution;
8. e-source: https://hi-news.ru/games/algoritm-google-obygral-60-igrokov-v-go-bez-edinogo-porazheniya.html;
9. Boston dynamic. E-source: http://www.bostondynamics.com/
10. Asilomar AI Principles. E-source: https://futureoflife.org/ai-principles/
11. Безумовний дохід. E-source: http://ua.euronews.com/2017/02/21/universal-income-a-game-changer-or-futile;
12. E-source: http://www.businessinsider.com/bank-of-america-wonders-about-the-matrix-2016-9?IR=T
13. Business Insider, http://www.businessinsider.com/cyborgs-are-the-future-of-financial-advice-2017-2;
14. https://www.wsj.com/articles/how-ai-is-transforming-the-workplace-1489371060
15. http://science.sciencemag.org/content/354/6314/900;
16. https://www.buzzfeed.com/venessawong/the-computerized-lettuce-factory-of-the-future?utm_term=.ksDgr2oOq#.rtwBvDLwn;
17. https://futureoflife.org/ai-principles/
18. Nick Bostrom. Superintelligence: Paths, Dangers, Strategies.// Oxford. – 2014. – 411p.

Authors' Information

Volodymyr Hrytsyk *– Ternopil Ivan Puluj National Technical University; professor of Software Engineering cathedra., Ruska str., 56, Ternopil, 46001, Ukraine; e-mail: volodymyrhrytsyk@gmail.com*

Major Fields of Scientific Research: Pattern recognition and Artificial Intelligence.

TABLE OF CONTENTS