DESIGN AND EVALUATION OF A WEB SYSTEM SUPPORTING VARIOUS TEXT MINING TASKS FOR THE PURPOSES OF EDUCATION AND RESEARCH

Karol FURDÍK*, Ján PARALIČ**, František BABIČ**, Peter BUTKA***, Peter BEDNÁR*
*Centre for Information Technologies, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 042 00 Košice, Slovakia, tel.: +421 55 602 4219, e-mail: karol.furdik@tuke.sk, peter.bednar@tuke.sk
**Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 042 00 Košice, Slovakia, tel.: +421 55 602 4220, e-mail: jan.paralic@tuke.sk, frantisek.babic@tuke.sk
***Department of Banking and Investment, Faculty of Economics, Technical University of Košice, B. Němcovej 32, 040 01 Košice, Slovakia, tel.: +421 55 602 4219, e-mail: peter.butka@tuke.sk

ABSTRACT

This paper presents an original solution that offers necessary functionalities for design, implementation or simple evaluation of various text mining techniques based on Java library called JBOWL. This library was designed as open source API to support different phases of the whole text mining process and offers a wide range of relevant classification and clustering algorithms. JBOWL is particularly useful for enhancing existing software applications with text mining capabilities, as well as for support of practical education of text mining and its exploitation. In this paper we present two particular cases where JBOWL has been successfully integrated and tailored for specific way of exploitation. First case presents integration of JBOWL within collaborative application called KP-Lab System and the second one is a web-based system for education purposes. The proposed solution supports the whole text mining process, starting from creation of a corpus of relevant documents, application of various pre-processing methods, up to creation of text mining models in a form of classifiers and evaluation of the obtained models. The execution of different tasks in the same time is supported by task-based execution engine, which provides middleware-like transparent layer for distributed execution. Evaluation of developed solution was realized within the university course called Knowledge management. This course is organized at the Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice. The paper also describes performed experiments and their results.

Keywords: *text mining, semantic annotation, natural language processing, corpus, experiments*

1. INTRODUCTION

The text mining is a process aimed at extraction of potentially interesting and useful information from large sets of texts. The whole process consists of several phases that can be supported by suitable software tools, but the feedback from users is always required. One possible exploitation domain of text mining can be e-learning in order to support management of various shared (learning) objects, mainly text files. The basic procedures contain upload, sharing and deletion, but some advanced methods are needed as possibility to create annotations, categorization into relevant categories, semantic analyses and some others. This set of features is covered by proposed solution called JBOWL application that was designed and is still being further extended as an open source with the intention to provide an easy extensible, modular framework for pre-processing, indexing and further exploration of large text collections.

The shared objects of interest are based on goals of user's activities and have to be available on users' requests. In order to correctly handle these requests, the objects need to be described by means of semantic metadata that typically consists of proper keywords, conceptual classification categories, and the properties as author, location, availability, creation date, etc. (see e.g. standards like Dublin Core[1], IEEE Learning Object Metadata[2], and similar). Relationships between individual objects are based on connection of relevant metadata to the semantic structures of an ontology-based knowledge representation.

Creation of a consistent metadata description for shared objects is not an easy task and may often be difficult for users. One of promising approaches is the use of text mining techniques for suggestion of a specific metadata description. This problem can be supported by the text mining techniques originating from two different groups [2], [3]:

- ***Classification*** (supervised learning) – is suitable for categorization of the objects into some predefined categories (e.g. concepts from an ontology).
- ***Clustering*** (unsupervised learning) – it is possible to search for clusters (groups) of similar shared objects.

Design and development of the proposed complex web system enhanced by the text mining facilities requires a competence and experience in the fields of knowledge management, information retrieval and natural language processing. It was accomplished by the involved team members of the Centre for Information Technologies (CIT)[3]. This centre takes part in several Slovak and European research projects with orientation on knowledge management in different domains. Some of the projects where people from CIT have been involved in are: Webocracy [11], KnowWeb [12] and currently KP-Lab[4].

KP-Lab (Knowledge practices laboratory) represents an European IST project which aims at developing theories, tools, and practices that significantly enhance understanding of knowledge creation processes as well as

[1] http://dublincore.org/
[2] http://ltsc.ieee.org/wg12/
[3] http://web.tuke.sk/fei-cit/index-a.html
[4] http://www.kp-lab.org/

ISSN 1335-8243

working practices [1]. The project consortium, which includes 22 partners from whole Europe and Israel, provides theoretical framework as a baseline for collaborative virtual system with many integrated functionalities to support collaborative knowledge creation processes. However, it was early recognized that the outcomes of the KP-Lab project would have a very limited application potential in Slovak conditions, mainly due to lack of a specific support for Slovak language.

To overcome this problem, the CIT team decided to initiate a new national APVV project *PoZnaŤ*. One of the main goals in the project PoZnaŤ, which is described in section 2 below, was to adjust the system for use in Slovak conditions with necessary improvements in existing text mining tools. Section 3 provides a detailed description of the proposed system architecture, together with the main components as JBOWL, task-based execution engine and user interface. In section 4, performed experiments are presented and obtained results are discussed. Finally, section 5 concludes the paper with a brief summary.

1.1. Motivation

Semantic description of various objects is important for categorization, proper search functionalities and whole view over a large set of data. The semantic description is represented by the metadata that can be created partially by users and partially also in an automatic manner. One type of the user-created metadata is that of semantic annotation, which can be used for detailed specification of relevant objects and further employed as a source for classification or clustering methods. These two methods can help to divide a possibly large dataset into suitable semantic categories.

The whole process, based on employing the text mining methods, includes the steps as acquisition of relevant documents, creation of the corpus of documents and identification of key terms in it. Further processing covers creation of a classification model for this corpus based on the identified key terms, division of corpus into training and testing datasets (in case of classification), creation of relevant classifiers (or clusters) based on training dataset, execution of proposed experiments on testing data based on learned classifiers, and evaluation of acquired results.

Text mining approach is relatively demanding computing process, especially if there are large datasets and many users. The interesting improvement in that case is the execution engine for text mining tasks that provides possibilities to run the tasks in parallel, in a distributed environment.

1.2. Related work

The decision to design and implement a tool for support of text mining and retrieval functionalities was based on the detailed analysis of existing free software tools that could be used to support the following functionality requirements [13]:

- Be able to efficiently pre-process potentially large collections of text documents with flexible set of available pre-processing techniques.
- Particular pre-processing techniques should be well adopted for various types and formats of text (e.g. plain text, HTML or XML).
- Text collections in different languages were envisaged, e.g. English and Slovak, as very different sorts of languages require significantly different approaches in pre-processing phase.
- Support for indexing and retrieval in these text collections (and experiments with various extended retrieval techniques).
- Well-designed interface to knowledge structures such as ontologies, controlled vocabularies or WordNet.

We found four different categories of tools:

- Text indexing and retrieval tools (such as e.g. Lucene[5]),
- Tools for text processing (e.g. GATE [14], JavaNLP[6]),
- Tools and APIs for support of the process of knowledge discovery in databases (Weka [15], KDD Package [16], JDM API [17]),
- Frameworks for work with ontologies (e.g. KAON [18]).

Features of all categories are summarized as follows:

- Text analysis is supported in Lucene, GATE, JavaNLP, KAON and Weka.
- Vector representation is provided in different ways from no support in Lucene, through base support in GATE, JavaNLP, KAON, to not optimized in KDD Package, JDM API and Weka.
- Mining models can be realized through KDD Package, Weka, JDM API and GATE (text extraction).
- GATE and KAON provide interfaces for work with ontologies.
- Full-text search is implemented only in Lucene.
- NLP methods are implemented in GATE and JavaNLP.

Each of the mentioned applications covers several described requirements, but none of them can be marked as fully suitable solution for text mining and semantic retrieval, e.g. Gate is strong oriented on pre-processing phase and has low support for text mining algorithms and their modifications. On the other hand, Weka provides simple pre-processing methods with the need for transformation of the text files into Weka internal format arff [19], which is not suitable for text mining analysis with sparse matrices. There are also some other extensions for Weka, e.g. commercial tool by AINetSolutions [20] that allows recursive indexation of documents that are stored in directories and later to transform the inverse index to a direct index in Weka format (arff) assigning to every document a certain category. The important fact is that all of them are still in development based on new conditions and requirements for adaptation to them.

Proposed solution for support of various text mining techniques provides an easy extensible and easy to learn

[5] http://lucene.apache.org
[6] http://nlp.stanford.edu/javanlp/

modular framework for preprocessing and indexing of large text collections, as well as for creation and evaluation of supervised and unsupervised text-mining models through simple user environment.

2. PROJECT POZNAŤ

Project PoZnaŤ[7] aims at development of a suite of software tools, techniques, and data repositories for processing of Slovak language (but to leave it open also for another languages), then an adaptation of the tools on extraction of knowledge from Slovak texts and documents, and finally a verification of system functionality on the pilot application. These project objectives have been accomplished through several specific activities as follows.

The first specific activity was based on the development of an integrated suite of tools for NLP in Slovak language. It contains transformation of existing tools developed within KP-Lab project to provide means to process the textual and multimedia documents written in Slovak language, to extract knowledge fragments from them, to integrate them and form them into resulting knowledge.

Design and development of the system components was driven by a division into particular language levels, e.g. morphology, derivatology (word-formation), syntax (both deep and surface), and semantics, with relations to the existing structure of knowledge representation. Particular tools were designed as accessible via web service interface and were integrated into a specialized web portal[8].

The second activity resulted in the design of a data repository within a CMS solution based on Content Repository API for Java standard (JCR, also known as JSR 170)[9], using the Jackrabbit implementation[10]. The data repository is a core of the proposed system and consists of a corpus of training texts accompanied with the data structures needed for particular phases of processing on the language levels. Several resources are already available and are envisioned to be adapted and integrated into a common format (based on XML, compatible with standards recommended in the field of corpus linguistics [10]).

The last specific activity represents evaluation of the project outcomes within a pilot application. Particular software components as well as the integrated system itself, have been verified and evaluated on the pilot application within the course of Knowledge Management[11] held on the Department of Cybernetics and Artificial Intelligence, at the Technical University of Košice. During this course, students were asked to accomplish predefined text mining tasks in the implemented web application (cf. Section 3.4). Besides testing of the core system functionality, the multi-threaded and distributed modes of the web application were also evaluated.

[7] http://web.tuke.sk/fei-cit/poznat/index-a.html
[8] http://cit.fei.tuke.sk:8080/textminingweb/
[9] http://www.jcp.org/en/jsr/detail?id=170
[10] http://jackrabbit.apache.org/
[11] http://people.tuke.sk/jan.paralic/mz.html

3. PROPOSED SYSTEM

The architecture of the proposed system is based on three main sets of tools, see Fig. 1:

- Tools and services developed within the KP-Lab project, adapted on new conditions and requirements;
- Existing tools and services provided by JBOWL library;
- Newly proposed and implemented tools and services.

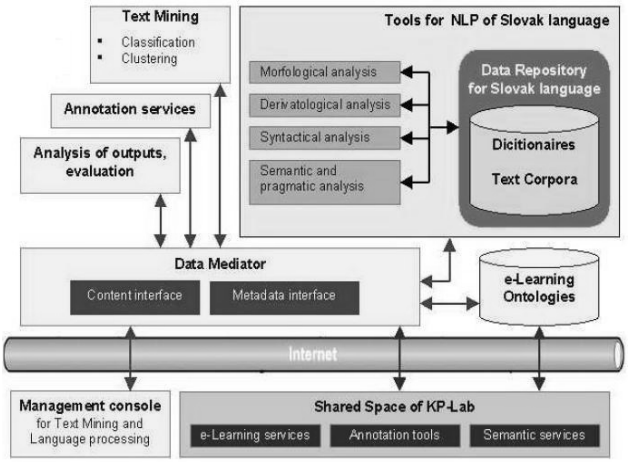

Fig. 1 Architecture of proposed application

The system based on presented architecture was designed to provide support for user activities within text mining process (from preprocessing to evaluation phase). Students interact with this system using simple web-based environment with necessary integrated functionalities. These activities allow them to design and execute their own text mining experiments based on theoretical knowledge and acquire practical experiences.

3.1. Architecture

The "Shared Space of KP-Lab" box in Fig. 1 represents components that were implemented within the KP-Lab project and are used in its original form. The "Tools for NLP of Slovak language" box represents features, components and modules that were designed and actually developed within the PoZnaŤ project. The rest of architecture components cover original KP-Lab services that have been reused and modified within the PoZnaŤ.

The core of designed application is the ontology-based semantic representation of e-Learning entities and processes. Both types of entities, i.e. shared objects and activities, are semantically described, i.e. annotated, by the metadata – ontology concepts. Process of identification, creation, maintenance, and usage of the semantic metadata depends on the content (meaning) of shared objects and actions, which can be extracted from the textual description.

The textual description of the shared objects and user actions is language-dependent. That's why it was necessary to use NLP tools for processing the entities described in Slovak language. To create adequate semantic metadata descriptions, the NLP tools were

required to provide a good quality of meaning extraction from texts, as well as proper level of automation, with minimum of required administrative interventions.

In this process, and also during the qualitative evaluation of processes, the mechanisms of text mining (classification, clustering, and others) will be applied. Promising results can be achieved by combining text mining with NLP methods.

3.2. JBOWL

JBOWL[12] is a Java library that was designed to support different phases of the whole text mining process and offers a wide range of relevant classification and clustering algorithms. This library was proposed as an outcome of the detailed analysis of existing free software tools in the relevant domain [2].

The initial set of JBOWL functions, originally developed by P. Bednár, is continuously being extended and improved, based on new requirements or expectations expressed by researchers and students of the Dept. of Cybernetics and AI, as well as of the Centre for Information Technologies. Currently, JBOWL provides an interesting open source solution in the domain of text mining for education or research purposes. The main problem for broader use of the JBOWL by users without programming skills was lacking graphical user interface. Within the project PoZnaŤ, the web-based user interface was designed and implemented based on Java standards and JBOWL itself has been improved in several ways. Namely, the execution engine supporting distributed processing of text mining tasks was implemented and the library was integrated with a suitable CMS solution based on existing Java standards as JCR 170, etc.

Detailed information about JBOWL library, its architecture, components and implemented features can be found in [4], [5] and [6].

3.3. Task-based execution engine

One type of the experiments performed within JBOWL was running of text mining tasks in a distributed environment, using the grid architecture. It means that whole text mining process, classification or clustering, was divided into several parts resulting in a service-oriented workflow composition tool [9]. The second motivating point in this situation was possibility to execute several tasks in the same time. This is an important feature especially for education purposes, because student groups can execute several sessions simultaneously.

Based on these new requirements, task-based execution engine was designed and has been implemented. This engine provides middleware-like transparent layer (mostly for programmers wishing to re-use functionality of the JBOWL package) for running of different tasks in a multi-threaded environment.

Detailed information about this feature was published in [4], [5] and [6].

3.4. Web user environment

The graphical user environment, also referenced as the text mining management console (see Fig. 2), has been designed and implemented as a standard JSP web application. The intention behind was to enable the users to access all the functionalities implemented in the JBOWL library, namely:

- Create, edit and delete a text mining project;
- Upload a new corpus of documents;
- Create and edit proposed categories for classification;
- Create semantic annotations of documents in corpus;
- Edit and delete of uploaded documents;
- Create an index from documents in corpus;
- Create a new classifier based on available classification algorithms and their parameters;
- Classify the corpus of documents using the created classifier;
- Evaluation of acquired results (documents are divided into relevant categories).

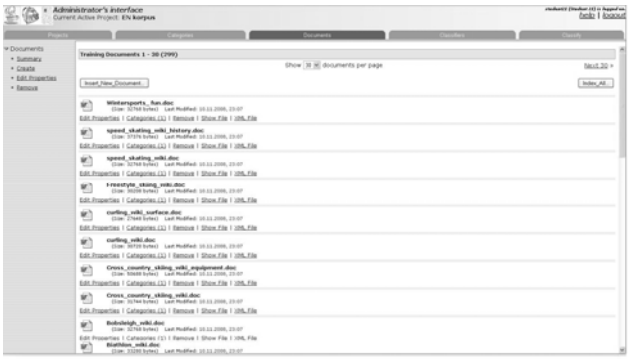

Fig. 2 Text mining management console with corpus of documents

To create a new classifier, the web application provides up to 10 different classification algorithms that can be further refined by several parameters (e.g. levels of weighting and normalization, way of stemming, stop-words extraction, etc.). For a given corpus of documents, the algorithm and its settings can be selected manually. In addition, a method for automatic selection of the most appropriate classification algorithm was originally designed within PoZnaŤ project. The method employs the MUDOF meta-learning approach [7]. This method was implemented into the JBOWL library and is also included in the web application interface.

4. EXPERIMENTS

Presented system is being continuously exploited for educational and experimental purposes within the education course called *Knowledge management* at the Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, TU of Košice. This course provides information about actual trends in the domain of knowledge management systems, including collaborative systems, information retrieval, text and web mining. Students obtain detailed information

[12] http://sourceforge.net/projects/jbowl

about knowledge representations, different types of processes, available algorithms and methods, etc.

The common definition for each exercise usually is:

- Identification of a concrete text mining task;
- Acquisition of relevant documents (corpus) and pre-processing based on selected algorithms;
- Creation of classification schema based on selected domain of documents;
- Division of the whole corpus into two sets, training set and testing set;
- Creation of new classifiers based on training set;
- Running the experiments on testing data;
- Evaluation of acquired results, precision of algorithms, finding a proper setting of algorithm parameters.

The following three cases have been selected for this paper and briefly described below in section 4.1-3. Their full documentation can be found on the wiki system of our course, which is available on http://kplab.fei.tuke.sk/mz/, but from obvious reasons in the Slovak language only.

4.1. Parallel classification of Slovak and English texts (case no. 1)

This exercise aims on dependencies between classification methods and language of used documents. Students prepared two corpuses from a single domain (wikipedia.org), one in Slovak and another in English language.

Both corpuses were created manually through download of relevant materials by members of the students' team. This approach was chosen in order to obtain balanced and representative corpus.

In parallel, students identified key categories in selected domains and started to create classification scheme. In this case, proposed scheme consisted of 4 main categories and about 30 subcategories.

Acquired corpuses contained documents in .doc format, mainly about 1-5 pages, each subcategory included about 10-20 documents.

Each corpus was divided into training and testing sets with a ration of 80% to 20%. This ratio was selected based on expectation to achieve better final classifier. The common decision in this situation is at least 60% documents in training sets.

The group of students performed some initial tests in the management console in order to get familiar with the user environment. They tried to make simple experiments based on the data that are part of the application.

First part of divided corpus was uploaded into the application to create a training set. This set was created together with its semantic annotation. Semantic annotations in this case categorize each uploaded document from training set in one of the defined categories. This categorization can be performed manually or automatically with usage of heuristic algorithms.

The group specified two basic evaluation criteria for evaluation of produced classifiers - recall (r) and precision (p).

$$r = \frac{tp}{tp + fn} \qquad p = \frac{tp}{tp + fp}$$

tp – true positive (correct classified documents for this category),
fp – false positive (documents that do not belong to this category, but were classified into it),
fn – false negative (documents that belong to this category, but were not classified into it).

Experiments contained several classification algorithms with different parameters' settings, e.g. term frequency, document frequency, tokenizer type, etc. The best classifier in this particular case was Support vector machine on the both corpuses (no. 3).

The whole experimental package contains 7 experiments with different classifiers. The experiments were executed on two corpuses, one in Slovak and other in English language. Precision in these two cases was about 80 %, but the value of recall was in average about 50 %. The best result was provided by classifier no. 3, precision (Sl)= 100%, precision (El)= 96%, recall (Sl)= 58% and recall (El)= 66% with the following parameters:

- Term frequency – normalized term frequency. It means that frequency of each term was divided by the number of occurrences of the most frequently term.
- Document frequency – without IDF. No inverse frequency was used in this case; there was no need to set up different weight for each term.
- Normalization – without normalization.

Described outputs of this case show that the best results were obtained with stemming (reducing words to their stem (root) form) based on Porter Stemmer and Support Vector Machine classification algorithm on corpus of text files in English language. Differences between Slovak and English text are caused by better realization of preprocessing method for English.

4.2. Classification of economic texts in Slovak language (case no. 2)

This exercise aimed at evaluation of the text mining system based on real experiments within group of students. The other main goal was to test suitability of proposed solution for Slovak language conditions.

In the initial phase, group of students obtained some information about process of text mining, detailed description of available algorithms, typical text mining tasks, etc.

The second step was to create taxonomy for domain of economics. Created classification scheme should represent main concepts for investigated domain, as microeconomics, macroeconomics, etc.

Corpus for experiments was generated based on created taxonomy and contained different files in pdf or txt format from internet (blogs, articles, etc). Final version of the corpus was evaluated as representative enough and well balanced, i.e. acquired documents represented the whole domain and the number of documents in each category was similar.

Training dataset was created as 2/3 of the whole corpus and provides categorization into 23 categories with about 10 documents in each category. Testing dataset contained 116 documents.

Experiments in this group were executed with selected algorithms, i.e. *perceptron*; *k-nearest neighbors*, *decision tree* and *decision rules*. Each method was described in details and this theoretical knowledge was used for setting up parameters of the relevant algorithm.

The second corpus was divided into two datasets, training and testing, in the rate of 70% and 30%. Required classifiers were created based on selected algorithm as *support vector machine*, *perceptron*, *k-nearest neighbors*, *boosting* and *decision tree*.

Selected algorithms were evaluated based on precision criterion only and obtained results were for particular classification algorithms the following: perceptron 45,7%, k-nearest neighbors 58,6%, decision tree 42,24% and decision rules 35.8%.

These results were not very optimistic. They can be improved through several actions as consultations with domain expert during creation of taxonomy, categorization of one document into several categories, improvements in the phase of pre-processing, and providing larger corpus of documents.

4.3. Evaluation of text mining application from users' point of view (case no. 3)

The goal of this evaluation was to identify positive and negative aspects of designed and implemented solution, to identify possible improvements in order to get the user environment simple and efficient enough.

The whole process consisted of several steps, starting with creation of documents' corpus. The first corpus was created in automatic way, i.e. students used web crawler called Web Harvest[13]. This first corpus was used for experimental evaluation of manual categorization directly in the web application based on semantic annotation. The second corpus contained annotated documents from New York Times web page that represent articles of this newsletter. Students have used it because all documents were categorized into defined categories with relevant annotations.

Manual categorization provides possibility to add category for each document individual based on user expectations. Users have to manually upload documents one after another and mark each document with relevant category. This approach is quite tedious for large set of data.

The last presented case was organized as usability study, so obtained results were used for proposal of necessary improvements in the proposed text mining system. Some suggestions for improvements that have been identified by students were, for example, a need of built-in feature for documents download based on URL of the source, possibility to categorize documents during upload phase, and more detailed documentation with examples.

[13] http://web-harvest.sourceforge.net/

4.4. Evaluation

The three described cases provide overall view of available functionalities implemented in proposed web system supporting the whole text mining process. The important fact is continuous development of this solution based on realized experiments within described pilot course, some bachelor, master or PhD. thesis. Each of them represents different utilization of core functionalities and their adaptation on existing conditions or expecting results. Results of these evaluation steps were used for necessary modifications and improvements that resulted in current version of the system. The main improvements are the following:

- Simple user web interface for accessing the JBOWL functionalities.
- Open source API for further usage of implementing new algorithms and methods.
- Task-based execution engine for possible execution of several tasks in the distributed way.
- Realization of data repository within a CMS solution based on Content Repository Java API.
- Possibility to upload a large set of documents with defined categories and saving them into implemented data repository.
- Creation of several different corpuses based on examined domains in various languages, mainly in Slovak and English.
- Preparation of detailed documentation about proposed solution that will be a part of published lecture notes for text mining. This initiative is very important, because many students have problems with practical realization of their theoretical knowledge about text mining process. They have a weak knowledge about the algorithms and their parameters that have to be set for successful realization and potentially useful results.
- Extension of initial set of algorithms with new implemented methods for classification, clustering, automatic selection of algorithms for classification and some others.
- Intensive dissemination of proposed solution to achieve its broader adaptation for education or research purposes.

5. CONCLUSIONS

Text mining provides possibility to analyze large sets of text files based on the selected techniques and application-dependent requirements of users. The whole process consists of the several phases that could be supported by different applications or tools. Our proposed solution covers all of these phases and provides possibility to bring required text mining capabilities into other applications based on the implemented features, together with free available API for the implementation of new algorithms and methods.

Actual version of the described solution was tested and evaluated with respect to usability and functional criteria and is continuously exploited in the education process

within the Knowledge management course of our Department. This version provides a web-based user environment with access to all functionalities provided by JBOWL and access to the content repository implemented through Jackrabbit open-source Java solution. Our solution offers wide scope of functionalities for realization of text-mining process within all its phases. As a result the proposed solution is suitable to support any suggested text-mining application designed by the users of JBOWL API.

One of the possible extensions of our system, already tested in experiments, is an integration of some heuristic algorithm based on the meta-learning method for automatic selection of algorithms in text classification task [7].

ACKNOWLEDGMENTS

The work presented in this paper was supported by the Slovak Research and Development Agency under the contracts No. RPEU-0011-06 (project PoZnaŤ) and No. VMSP-P-0048-09 (project DMM) by the Slovak Grant Agency of Ministry of Education and Academy of Science of the Slovak Republic within the projects No. 1/4074/07, No. 1/0131/09 and 1/0042/10; project implementation Centre of Information and Communication Technologies for Knowledge Systems (project number: 26220120020) supported by the Research & Development Operational Programme funded by the ERDF.

REFERENCES

[1] BABIČ, F. – BEDNÁR, P. – FURDÍK, K. – PARALIČ, J. – PARALIČ, M. – WAGNER, J.: Trialogical learning in practice. In: Acta Electrotechnica et Informatica, Vol. 8, No. 1 (2008), pp.32-38, ISSN 1335-8243.

[2] BEDNÁR, P. – BUTKA, P. – PARALIČ, J.: Java Library for Support of Text Mining and Retrieval, In: Proceedings of Znalosti 2005, Stará Lesná. VŠB TU Ostrava, 2005, pp. 162-169, ISBN 80-248-0755-6.

[3] BERRY, M. W.: Survey of Text Mining. Springer-Verlag, New York Inc., 2004. ISBN 0-387-95563-1.

[4] BUTKA, P. – BEDNÁR, P. – BABIČ, F.: Use of task-based text-mining execution engine in support of knowledge creation processes, In: Proc. Of Znalosti 2009 Brno, Czech republic. Vydavateľstvo STU, Bratislava, 2009, pp. 289-292, ISBN 978-80-227-3015-0.

[5] BUTKA, P. – BEDNÁR, P. – BABIČ, F. – FURDÍK, K. – PARALIČ, J.: Distributed task-based execution engine for support of text-mining processes, In: SAMI 2009: 7th International Symposium on Applied Machine Intelligence and Informatics: January 30-31, 2009, Herľany, Slovakia, IEEE, 2009, pp. 29-34, ISBN 978-1-4244-3802-0.

[6] BUTKA, P. – BEDNÁR, P.: Design and implementation of task-based middleware execution engine for JBOWL text-mining library, In: WDA 2008: Workshop on Data Analysis: Proceedings of the 8th International Student Workshop, Dedinky, Slovakia June 26-29, 2008, Košice: Equilibria, 2008, pp. 63-70, ISBN 978-80-89284-21-4.

[7] FURDÍK, K. – PARALIČ, J. – TUTOKY, G.: Meta-learning Method for Automatic Selection of Algorithms for Text Classification, In: Proc. of the Central European Conference on Information and Intelligent Systems (CECIIS 2008), 24-26 September 2008, Varaždin, Croatia, pp. 477-484, ISBN 978-953-6071-04-3.

[8] PARALIČ, J. – BABIČ, F. – WAGNER, J. – SIMONENKO, E. – SPYRATOS, N. – SUKIBUCHI, T.: Analyses of knowledge creation processes based on different types of monitored data, In: Proc. of the ISMIS 2009. Foundations of Intelligent Systems. LNCS 5722/2009, Springer Berlin / Heidelberg, pp. 321-330, ISBN 978-3-642-04124-2.

[9] SARNOVSKÝ, M. – PARALIČ, M.: Text Mining Workflow Construction With Support of Ontologies, In: Proc. of the 6th IEEE Int. Symposium on Applied Machine Intelligence 2008, Budapest Tech, 2008, pp. 131-135, ISBN 978-1-4244-2106-0.

[10] SNK: Slovak National Corpus. On-line: http://korpus.juls.savba.sk.

[11] PARALIČ, J. – SABOL, T. – MACH, M.: Knowledge Enhanced e-Government Portal. Proc. of the 4th IFIP International Working Conference on Knowledge Management in Electronic Government (KMGov 2003), Rhodes, Greece, May 2003, ISBN 3-540-40145-8, Lecture Notes in Artificial Intelligence 2645, subseries of LNCS, pp. 163 – 174, ISSN 0302-9743.

[12] Project KnowWeb home page, available on http://web.tuke.sk/kkui/projects/knowweb/KnowWeb.html.

[13] BEDNÁR, P. – BUTKA, P. – PARALIČ, J.: Java Library for Support of Text Mining and Retrieval. In: Proceedings of the 4th annual conference Znalosti 2005. Eds. L. Popelínský, M. Krátký. VŠB TU Ostrava 2005, pp. 162 – 169, ISBN 80-248-0755-6.

[14] CUNNINGHAM, H. – MAYNARD, D. – BONTCHEVA, K. – TABLAN, D.: GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02). Philadelphia, July 2002.

[15] WITTEN, I. H. – FRANK, E.: Data Mining: Practical machine learning tools with Java implementations, Morgan Kaufmann, San Francisco, 2000.

[16] BEDNÁR, P. – PARALIČ, J.: KDD Package. In Proc. of the Znalosti 2003 Conference, Ostrava,

February 2003, Czech Republic, pp. 113-122, ISBN 80-248-0229-5.

[17] JSR 247: Data Mining API 2.0, available on http://www.jcp.org/en/jsr/detail?id=247.

[18] STAAB, S. – STUDER, R.: An extensible ontology software environment, In Handbook on Ontologies, chapter III, pp. 311-333. Springer, 2004.

[19] Text categorisation with Weka, available on http://weka.wikispaces.com/Text+categorization+with+Weka.

[20] WekaIndex by AINetSolutions, available on http://www.ainetsolutions.com/eng/soluciones/aplicaciones/ir.html.

Received September 18, 2009, accepted February 11, 2010

BIOGRAPHIES

Karol Furdik received his Master degree in 1993 and his Ph.D. degree in 2003, both at the Technical University in Košice. Since 2007 he is working as a researcher in the Centre of Information Technologies, common workplace of Institute of Informatics, Slovak Academy of Sciences in Bratislava, and Technical University of Košice. His scientific research is focusing on the areas of natural language processing, text mining, knowledge management, semantic technologies, and eGovernment.

Ján Paralič received his Master degree in 1992 and his Ph.D. degree in 1998, both by the Technical University in Košice. Since 2004, he is associate professor at the Department of Cybernetics and Informatics, Technical University in Košice and since 2005 also head of the Centre for Information Technologies at the same university. He (co-)authored two books; (co-)edited 10 proceedings from various international workshops and conferences and published more than 70 scientific papers. His research interests currently are in the areas of knowledge discovery, text mining, semantic technologies, and knowledge management.

František Babič graduated (MSc.) at the Department of Cybernetics and Artificial Intelligence of the Faculty of Electrical Engineering and Informatics at Technical University of Košice in 2005. He finished the PhD. study at the same department in 2009. Since 2005 he is working as a researcher with the Department of Cybernetics and Artificial Intelligence in Centre of Information Technologies, Technical University of Košice. His scientific research is focusing on knowledge management, knowledge discovery and project management.

Peter Butka received his Master degree in 2003 at the Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics at Technical University in Košice. He studied artificial intelligence as a PhD student at the same department with thesis related to text-mining and semantic technologies. Since 2006 he is a researcher at the Faculty of Economics and also at the Department of Cybernetics and Artifical Intelligence. His scientific research is focusing on the areas of text mining, knowledge management, semantic technologies, and information retrieval.

Peter Bednár received his Master degree in 2001 by the Technical University in Košice. Since 2005 he is working as a researcher in the Centre of Information Technologies, common workplace of Institute of Informatics, Slovak Academy of Sciences in Bratislava, and Technical University of Košice. His scientific research is focusing on the areas of text mining, knowledge management, semantic technologies, programming (Java), eGovernment, and knowledge discovery.