# FUNCTIONAL MODULES OF A SEMI-AUTOMATIC QUESTION GENERATING SYSTEM

László BEDNARIK
Department of Information Technology, Comenius Faculty, Eszterházy Károly College, Hungary
Eötvös 7, 3950 Sárospatak, tel. +36 4751 3000, e-mail: bednarik@ekfck.hu

**ABSTRACT**

*In the last decade different applications of computer science (data mining, text mining) have been recognised. Presently an automatic question generation is a current research area. The elaborated system generates multi-choice questions from textbooks without using an external semantic database. Main demands made for the implemented automatic question generation system: supporting the Hungarian language, using a free grammar analytic, generating test questions from annotated texts, implementing potential answers from an internal, own developed semantic data base, supporting multi-choice problems, a possibility of implementing electronic and printable test papers. During planning the framework among the applied methods, clustering on the base of distances, a classification algorithm on the base of a neural net, in the case of generating questions and answer alternatives an own developed semantic data base was implemented. To prove the practical applicability of the automated question generating method, Java-language software was developed.*

***Keywords:*** *pre-processing, annotation, stemming, clustering, classification, multi-choice questions.*

## 1. INTRODUCTION

One of the future directions of the development of educational tools supported by informatics is signified by the adaptive and semantic oriented computer tutorial systems. Nowadays the main feature of architecture is the appearance of semantic databases storing the knowledge of the target topic next to traditional frameworks. Building the learning materials of these curriculum databases should provide high level of flexibility. The adaptability primarily means the capability to adapt to students' requirements. The flexibility appears in several functions of the system even in the knowledge testing module being the object for the study.

The created semi-automatic question generating system (AQG) can generate automatic multi-choice questions according to optional sentence types from annotated Hungarian text documents.

## 2. BACKGROUND

Research in connection with automatic question generation started in the 1990-ies. Earlier researches focused on the semantic aspect of question generation which serves as a methodological base of the present automatic systems.

The dominant approach in the research of creating AQG was recommended by Miller in 1995 [1], in which, based on the WordNet knowledge base six types of questions can be distinguished: definition, synonym, antonym, hipernym, hyponym and multi-choice. In 2004 Sumita proposed an automatic generation method for generating multi-choice questions [2], in which choosing sentences, defining blank spaces and incomplete sentences is carried out with computer learning methods.

Gütl et al. [3] created an automatic question generation system consisting of a module in English and German in 2008. On the base of the results of the developed prototype they improved the system which by now consisted of three modules: modules for pre-processing, extracting and generating questions [4].

## 3. FUNCTIONAL SYSTEM OF THE QUESTION GENERATION SYSTEM

In scientific literature most of the automatic question generation systems have been implemented to work with texts written in English and German. As working with text documents in Hungarian are still in an experimental stage an own developed system adaptable even in practice has been implemented.

### 3.1. Module for pre-processing text documents

The first important step of the question generating prototype system is the pre-processing whose main aim is to bring the documents into a form in which classification, clustering tasks can be carried out efficiently. The aim of the annotation module is to assign a role to each sentence. The annotation was carried out manually in the first phase of the prototype system. Currently, the goal of next phase is to automate the sentence annotation process. The annotation language was created as an enhancement of the descriptive scheme language DITA XML.

The module doing the pre-processing work supports filtering sentences in a great amount of text documents as input files implemented by annotation or classifying sentences into categories (concept, definition, declarative sentence). The pre-processing module of the implemented model can operate documents coded in formats DOC, DOCX, RTF, HTML and PDF as input data.

### 3.2. Stemming module

In scientific literature the procedure determining the stem of a given word is the aim is called stemming [5]. Through stemming, the set of words to be recognized and handled can be considerably reduced as in the Hungarian

language a basic word may occur in 20-50 inflected forms in the text. A free module the Szószablya framework [6] was built into the system whose algorithm works with the adaptation of the well-known Porter algorithm [7]. The main advantage of this method is the high speed.

Features are produced from analysed words in the text file separated by semicolon as follows: word form, stem, frequency, number of syllables, analysis, and part of speech.

### 3.3. Clustering module

The aim of clustering is to create separated groups from the object such that elements in a group should be as similar as possible and elements in separate groups should be as different as possible [8]. Clustering is a task of vital importance during automatic question generation. The clustering module waits for the set of words to be clustered as input information in order to operate. This set is produced by preceding modules filtering both sentences without annotation from a document used for question generation, and words in the list of stop words. The set of disordered words remained in this way provides one of the inputs for the clustering module. When it is ready according to an expert's opinion, a method to apply for word-distance configuration is chosen. The clustering module is represented in Fig. 1.

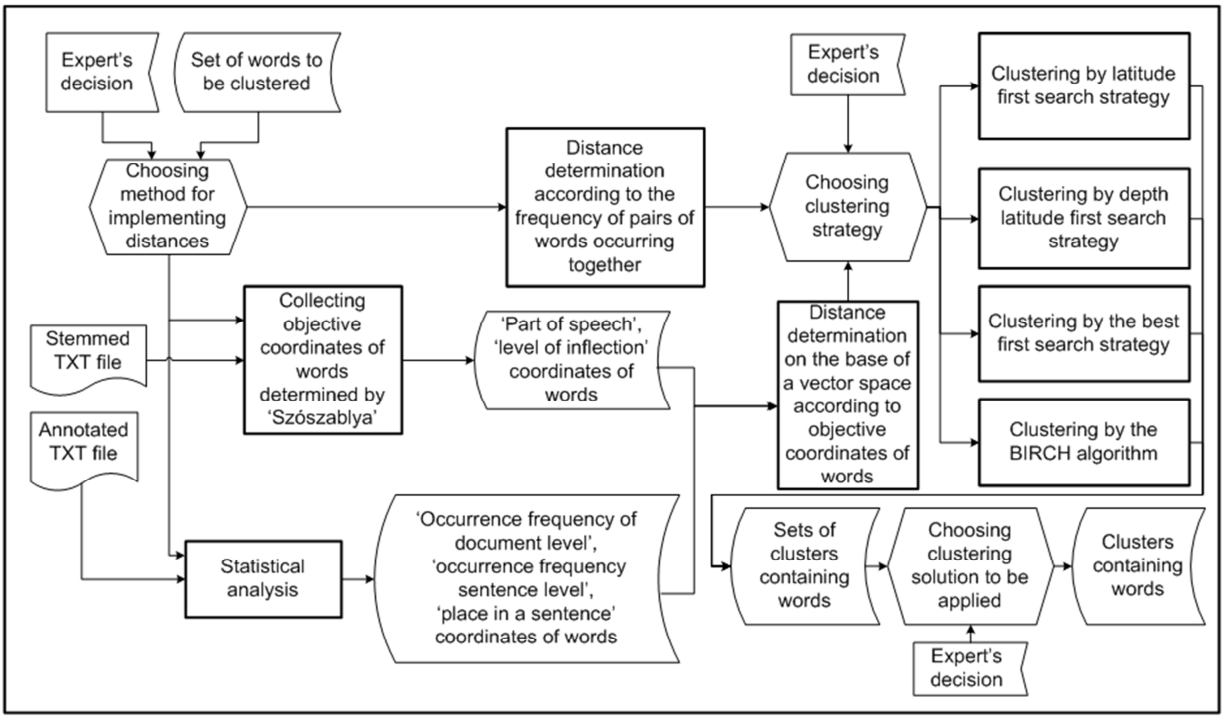


**Fig. 1** Clustering module

Clustering according to two concepts for distance configuration was implemented. In the first distance configuration method a similarity of two words was defined by the number of sentences in which two words occurred together in a document [9]

$$S_{i,j} = f_{ij} / \max(f_i, f_j) \quad (1)$$

where: $S_{i,j}$ is a number in the interval [0, 1] representing the distance correlation of words $i$ and $j$; $f_{ij}$ is the number of sentences in a document in which both words $i$ and $j$ occur; $f_i$ is the number of sentences in a document in which word $i$ occur; $f_j$ is the number of sentences in which word $j$ occur.

In the other representation distances of words were defined by their distances measured in a several dimensional space on the base of vector spaces defined by coordinates determined by objective methods.

Linguistic features objectively measurable by statistical methods of words are called objective coordinates. The objective coordinates used are: part of speech, measure of inflection, number of occurrence in the document, number of occurrence in the sentence, place in the sentence. On one hand information was produced by the stemming framework while on the other hand by my own statistical text analysing algorithm.

In this application four clustering strategies were worked out each of which represents a potential execution alternative of the clustering concept Hierarchical Agglomerative Clustering (HAC).

Beyond latitude, abysmal and the best first searches known from scientific literature we implemented the process of clustering by a less known algorithm Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) that yields a much more effective result than

other methods in the case of a task containing a considerable amount of elements to be clustered.

### 3.4. Classification module

The aim of the module is to define subjective coordinates linked to words according to coordinates of words measured objectively in a document. The neural net was taught according to a supervised teaching method. The goal of classification is to define the function describing the unknown relation $f : O \to 2^C$ as well as to determine a classifying function

$$g : O \to 2^C \tag{2}$$

for which

$$E(f, g, S) \to \min \tag{3}$$

holds where the error function is denoted by $E$, the function describing the unknown relation is $f$, the classifying function is $g$, the teaching set is $S$, and the set of codes is $C$. The function $g$ obtained in this way can be used to converge to the function $f$ on the whole set $O$. The classifying module is represented in Fig. 2.

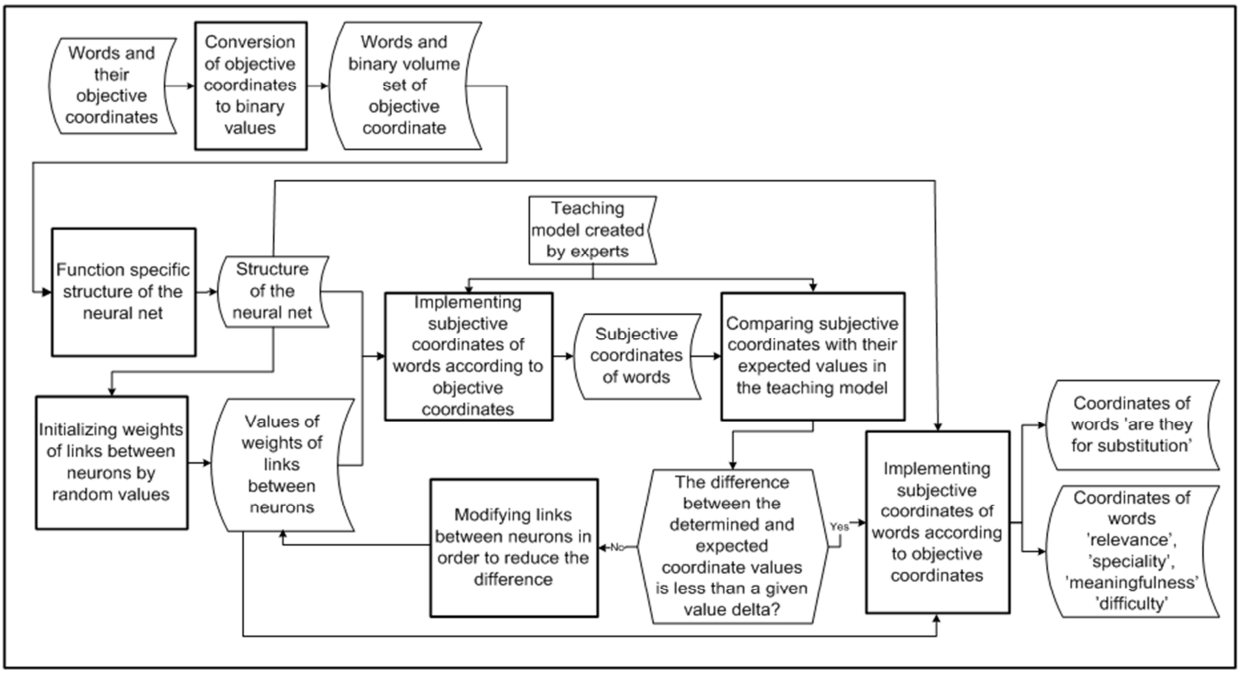


**Fig. 2** Classifying module

The module doing the classification gets words and objective coordinates linked to them implemented by the preceding modules as input information. Objective coordinates of words partly coincide with the objective coordinates produced by the clustering module, however, before the classification these coordinates are complemented by newer information gained by knowing the result of clustering. These are the new objective dimensions: the serial number of a cluster containing the word, moreover, the average distance of the word from the other words of the sentence containing the word.

Together with these in the input of classification each word is placed in a seven dimensional objective space. the represented neural net each neuron was modelled by an activation function of two values. The structure of the neural net is built up after learning the objective coordinates. The reason for this is that each classification task demands a uniquely built neural net, as the set of values of most of the objective coordinates is learnt only after clustering therefore the number of neurons competent to accept and forward input values can also be determined by learning the precise cardinality.

In order to solve this task a neural system of three layers with forwards links was planned where the number of neurons in the hidden layer was chosen identical to the number of neurons in the output layer. Before the classification the expert can teach the neural net according to the teaching model she/he implements. The teaching model contains the objective and subjective coordinates of an arbitrary number of words in an ordered form. Subjective coordinates denote features of words cannot be correctly determined and they reveal people's judgement concerning a given word. In the developed model system subjective coordinates are implemented as follows: relevance, speciality, meaningfulness, difficulty. Dimensions of the subjective space were determined such that features of words by human interpretation would be more emphasized and to provide valid support to decide on which words in a document should be extracted for a question generation. This procedure is executed

automatically by the classification algorithm.

As the first step of learning the weight values of the neural net are determined randomly on the interval defined for the weights according to uniform distribution. Afterwards according to the coordinates given by the expert together with the neural net, subjective coordinates of words are also determined. After determining subjective coordinates teaching algorithm evaluates the answer is correct according to subjective coordinates given by the expert. If the difference is greater than a pre-defined value the algorithm changes the weight values of links between neurons such that the difference from subjective coordinates given by the expert would decrease. According to the new weight values subjective coordinates of words in the teaching model are repeatedly determined.

The learning process is going on as long as the subjective coordinates determined by the neural net get adequately close to their values defined by the expert. After learning the objective coordinates of words to be classified get into the input of the neural net the answer to which will mean subjective coordinates of words. As the set of values of most of the subjective coordinates contains more than two values therefore conversion between values is necessary even in the output. The direction of it is opposite to the one applied in the input link. Here binary symbols produced by the neural net are converted such that they would gain several values. Among subjective coordinates the value of the coordinate named 'whether to be extracted as a question' is used to decide whether a word under examination is to be extracted as a question from a sentence. However, this coordinate plays a significant role. The exact decision is made by the module generating questions.

To assess the efficiency of classification three tests were made consisting of 30, 40 and 50 questions. The best results were obtained during testing with the test paper containing 30 questions. Hence the value of sensibility is 98.1%, in which the system considered true statements as false only in 1.9 % of all cases. The value of precision is 98.72% which is also the greatest in this type where the system considered false statements as true only in 1.28% of all cases. The value of the F-measure is 98.41% where the weight value of the sensibility and the precision are considered in the case of $\alpha = 0.5$.

### 3.5. Question and answer generating module

In this module sentences extracted as questions and words extracted from sentences as questions are defined. This module generates the optional answers to questions posed. The module generating questions and answers is represented in Fig. 3.

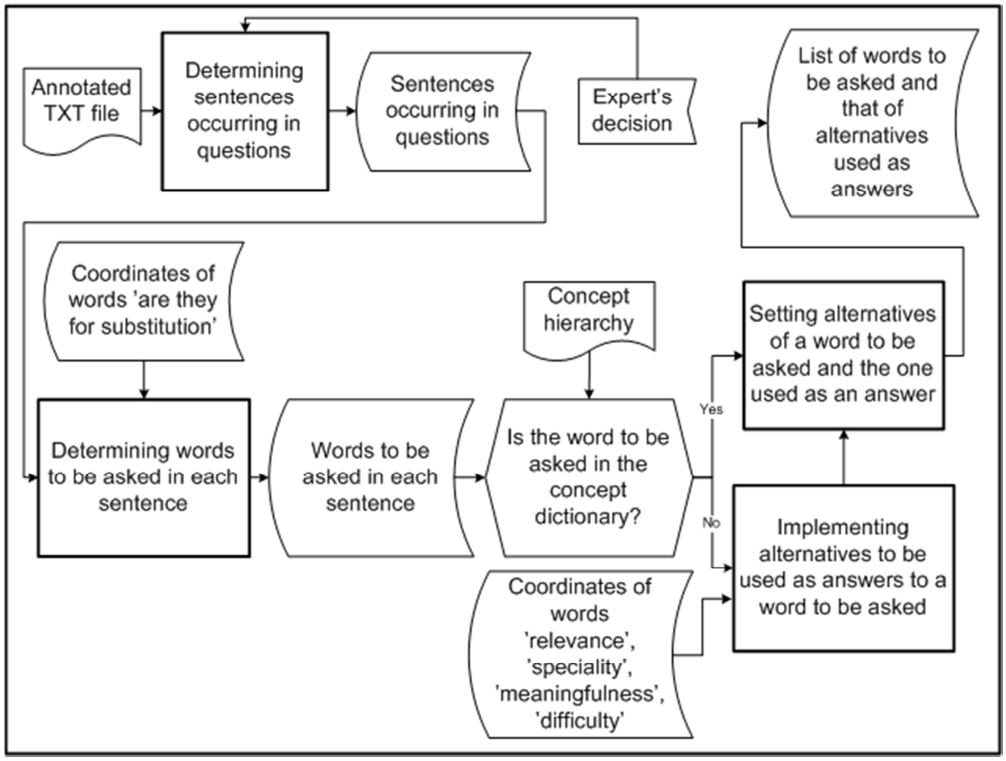


**Fig. 3** Question and answer generating module

As the first step in a question generation, sentences extracted as questions from the document are needed to be defined. In order to do it the module needs to get the text file containing sentences of the document annotated as input information. Afterwards according to an expert's decision types of sentences from which questions can be generated , may be determined. Apart from the type of sentences the number of sentences extracted as questions can be determined at this stage. It also happens according to an expert's opinion.

After determining sentences words that can be extracted as questions are chosen. Thereunto subjective coordinates namely 'whether to be extracted as a question' of words gained from the classification module make for

an orientation. As these coordinates characterize each word uniquely it may occur that several words in a sentence can be marked out for being extracted as a question. In this case the module ranks the possible alternatives according to the input function of the output neurons of the neural net. Words in which this value is greater are more likely to be extracted as questions. As the last step in the question generation, possible alternatives that can be given as answers are needed to be determined. The word extracted as a question has to be in one of them as a solution as well as optional answers being more and further away from it with respect to objective and subjective features. These alternatives being more and further are suitable to score the knowledge of a respondent. Modelling distances between alternatives was implemented according to several aspects. On one hand the method was used to model distances chosen in the clustering module and on the other hand we also planned it for the distance that can be determined by the relation between a word and a class word represented by a concept hierarchy. In that model the algorithm offers 5 potential alternatives for each word extracted as a question. The answer alternatives produced by the algorithm are revealed in a jumbled order for the respondent.

The question- and answer generating module work entirely automatically. The expert has the opportunity to change the setting of the programme before and after running the algorithm. Like choosing clustering strategy, number and type of questions, determining the electronic or printable form of a test paper.

## 4. IMPLEMENTATION OF THE AQG SYSTEM

A system for implementing an automatic question generation was developed such that a joint and aligned work of several data link subsystems ensures solving the problem. An input and output interface for each module was worked out. Input interfaces define data needed for modules to do their work while output interfaces define data modules have to provide. Standard model elements were used in creating the framework: data in text format, or rather data stored in a binary file, data stored in internal data representation, well-defined problem solving and decision making modules.
The developed question generation framework is shown in Fig. 4 [10].

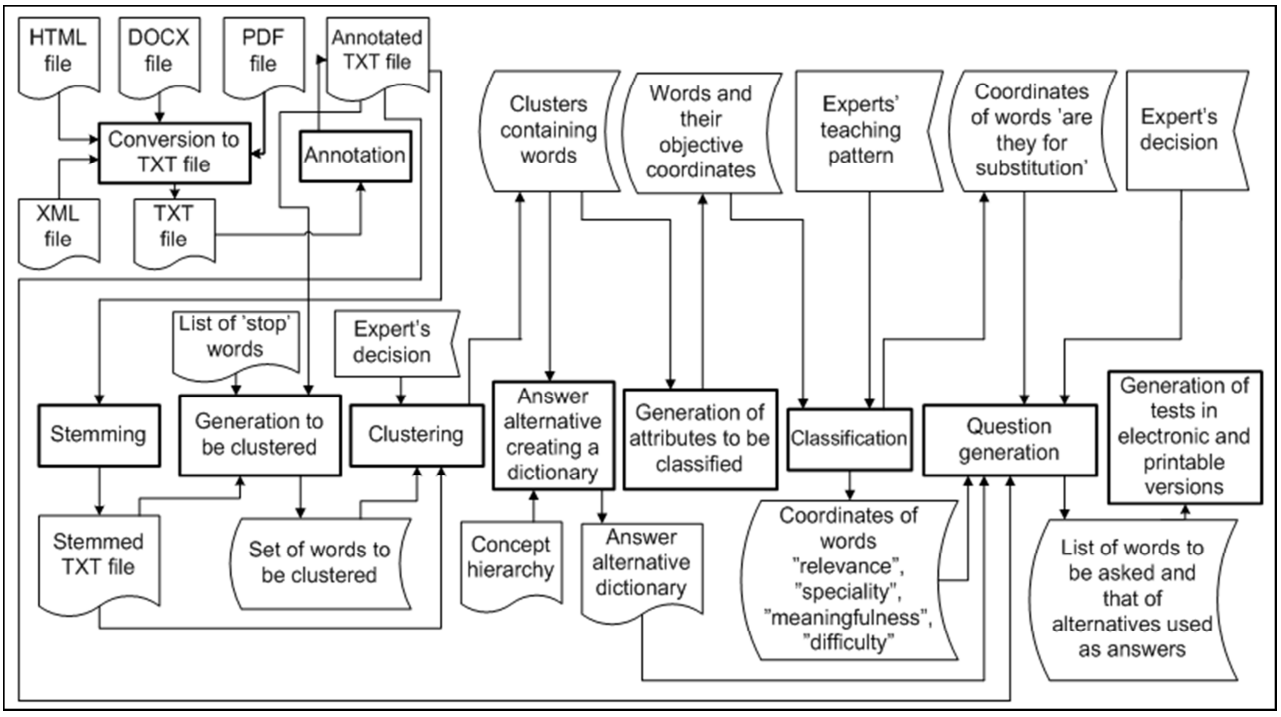


**Fig. 4** Modules of the developed question generation framework system

The correctness of the model was tested by software implemented in Java language. The developed software outputs questions in electronic or printable format. In the case of electronic version both the generated questions and possible answers appear in computer environment in front of the user. In this version, users attending the test can have a local menu on the screen offering all the possible answers to questions by clicking on the blank part of the sentence containing the question to be answered. After giving the answer the chosen alternative is automatically substituted in the sentence. After finishing the test, the filled test sheet can be saved in a file format and can be forwarded to the reviewer of the test. For representing the test sheet in a printable format the software indicates the place (blank part) of the word taken out as a question with dots for the person who is filling it in and the possible answers appear listed next to each other below sentences.

In order to fill the test in a printed format the user has to underline the word judged to be appropriate or write it where the dots are. Fig. 5 shows an example for the test sheet in the form it appears.

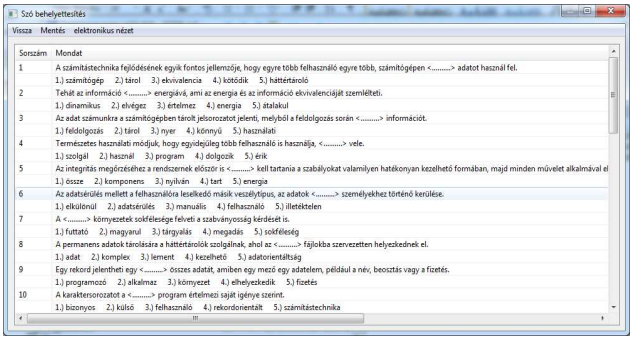


**Fig. 5** Printable version of the automatically generated test sheet

Most important differences between the implemented semi-automatic question generation system and the system EAQC developed by Gütl et al. (2011) are as follows. Annotated documents written in Hungarian are needed for the question generation. In order to solve the problem an application carrying out grammatical analysis is needed that determines the basic linguistic features of words in a document. An own developed dictionary of concepts was created for the system to work since the lexical knowledge base WordNet [11] was not available in the chosen area and language.

## 5. CONCLUSIONS

The prime aim of the research is to design a flexible and open framework, to implement it and to prove that it works in practice using the Hungarian language to generate questions and answer alternatives.

Several important requirements were considered while planning the implemented semi-automatic question generation model system. Among them we emphasize algorithms we developed and their optimisation means not only a decrease in the storage place but also an increase in the speed of creating the generated question and answer alternatives. In the present form this model system can analyse and process annotated text documents. The automation of the pre-processing module of the model system completely can be implemented by converting a text document in a document format DITA XML. It implies further research.

## REFERENCES

[1] MILLER, G.: WordNet, a lexical database for English, Communication of the ACM, vol. 38, 1995, pp. 39-41.

[2] SUMITA, E. – SUGAYA, F. – YAMAMOTO, S.: Automatic Generation Method of a Fill-in-the-blank Question for Measuring English Proficiency, Technical report of IEICE, 104 (503), 2004, pp. 17-22.

[3] GÜTL, C.: Automatic Limited-Choice and Completion Test Creation, Assessment and Feedback in modern
Learning Processes, Paper read at LRN Conference 2008, Guatemala, February 12th – 16th.

[4] GÜTL, C. – LANKMAYR, K. – WEINHOFER, J. – HÖFLER, M.: Enhanced Automatic Question Creator – EAQC: Concept, Development and Evaluation of an Automatic Test Item Creation Tool to Foster Modern e-Education, The Electronic Journal of e-Learning volume 9 Issue 1, ISSN 1479-4403, 2011, pp. 23-38.

[5] ATWELL, E. – DEMETRIOU, G. – HUGHES, J. – SCHIFFRIN, A. – SOUTER, C. – WILCOCK, S.: A comparative evaluation of modern English corpus grammatical annotation schemes. ICAME Journal, vol. 24, 2000, pp. 7-23.

[6] NÉMETH, L.: A Szószablya fejlesztés, 2003, pp. 3-4.

[7] PORTER, M. F.: Snowball, A Language for Stemming Algorithms, 2001.

[8] BODON, F.: Adatbányászati algoritmusok, Free Software Foundation által kiadott GNU Free Documentation license 1.2-es, Budapest, 2006, pp. 145-146.

[9] KOVACS, L. – BEDNARIK, L.**:** Application of Semantic Clustering in Question Generation Engine, Scientific Bulletin of the "Politehnica" University of Timişoara, Romania, Transactions on Automatic Control and Computer Science, vol. 57(1), No. 4, December 2012, ISSN 1224-600X, pp. 211-218.

[10] BEDNARIK, L.: Automatizált kérdésgenerálás annotált szövegből, Hatvany József Informatikai Tudományok Doktori Iskola, Miskolc, 2012, pp. 24-30.

[11] WordNet: A lexical database for English, Princeton University, (2010), http://wordnet.princeton.edu.

Received January 26, 2014, accepted March 6, 2014

## BIOGRAPHY

**László Bednarik** was born in Vajdácska, Hungary in 1957. He graduated from the University of Miskolc in 2003. He received Ph.D. degree in Applied Computational Science University of Miskolc in 2012. He works as an assistant professor at the Eszterházy Károly College, Faculty Comenius at the Department of Information Technology. He is author and co-author of more than 25 scientific papers. His research interest includes data and knowledge bases and knowledge intensive systems, applied information science as database management and artificial intelligence methods, as well as information management