Automatic Question Generation Based on Sentence Structure Analysis

Miroslav Blšták*
Institute of Informatics, Information Systems and Software Engineering
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava
Ilkovičova 2, 842 16 Bratislava, Slovakia
miroslav.blstak@stuba.sk

Abstract

Verification of knowledge in forms of questions belongs to the most important parts of educational process. But the creation of questions is difficult and time-consuming activity. With an increasing impact of interactive information technologies in the learning environment, it is also interesting to automatize the process of knowledge verification. Therefore, interest in automatic question generation (AQG) task is rapidly increases in the recent years. In this paper, we present our research findings and contribution to the AQG as a task of natural language processing (NLP). Whereas questions generation from text is a very wide area, we focus on factual questions with the purpose to verify user knowledge or her/his reading comprehension. There are various approaches to this task. After our deep analysis of current systems and their shortcomings, we decided to create question generation method combining traditional linguistic approach based on sentence structure analysis with machine learning methods. Creation of patterns for question generation process requires human experts, so we adopt data-driven approach to learn question generation steps automatically from initial set of sentence-question pairs. Our results confirm, that it is possible to train the algorithm for this task and our proposed method is also adaptable for related task of NLP field. We have used it also in sentence simplification step when we divided complex sentences into simpler ones. Except for the proposed method, other outputs of our work are datasets of sentence-questions pairs and question generation framework providing the possibility of evaluation of generated questions.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing; K.3.1 [**Computers and Education**]: Computer Uses in Education

Keywords

automatic question generation, natural language processing, machine learning, natural language generation, natural language understanding

1. Introduction

Questions are important part of our everyday life and an inseparable part of education. With the arrival of Web 2.0 and e-learning, educational encyclopedias based on collaboration and content expansion take the position in the online educational world. Many systems have been developed to support and manage learning content like learning management systems (LMS) or massive online open courses (MOOC). In parallel, alternative learning approaches have emerged and several of them are based on questions. For example, Self-Organized Learning Environment (SOLE), in which the whole education is done by asking questions [1]. So, it would be useful to build some type of an agent, which will be able to replace a human expert (teacher/lecturer) in this role.

In addition, education is not the only area where the question can be used. AQG systems are also used as an auxiliary module for question answering task [10, 12], for building databases of frequently-asked question section [16], for finding out if lectures are understandable [9] or for creating a dialog system in role of an interactive agent [6].

When we look at AQG as a task of NLP, it covers both directions of language processing which puts it among the most demanding task of this area (like tasks: text simplification or machine translation). Firstly, it is required to transform text as a set of characters into computer-understandable representation (natural language understanding), and consequently it is possible to create questions and represent them in form of text (natural language generation). The input for AQG task is a set of sentences (e.g. paragraph or article) and the output is a set of questions related to input sentences.

The importance of research in this field is also confirmed by The Question Generation Shared Task and Evaluation Challenge (QGSTEC) which has triggered interest in this task [13]. In this event, there was also created the

*Recommended by thesis supervisor: Assoc. Prof. Viera Rozinajová
To be defended at Faculty of Informatics and Information Technologies, Slovak University of Technology in Bratislava on [To be specified later], 2018.

first question generation dataset and participants of this event have competed in two tasks: question generation from sentences and question generation from paragraph. In combination with increasing significance of NLP development, the AQG systems have become very useful applications in our life.

The structure of this extended abstract is as follows. In the second section, we will briefly introduce drawbacks of the state-of-the-art systems as open problems in this area and we will establish goals of our work. In the third section, we will propose our question generation approach and prototype of our framework. In the fourth section, we will describe our experiments and in the last fifth section is the conclusion.

2. Open Problems and Dissertation Goals

We have done detailed analysis of AQG task and have identified several drawbacks of the state-of-the-art approaches and systems (e.g. [5, 6, 7, 8, 14] and many others). Most of the current systems leverages lexical, syntactic or semantic analysis of input text and then they construct questions based on hand-written patterns or rules. There are some problems based on the nature of this approach. The first constraint is, that we need some human experts (programmers or linguists), who create and manage the list of transformation rules. The second constraint lies in management of the transformation rules itself. When the rules are too general (usually constructed by combination of syntactic patterns with named entity identification), they cover many sentences, but with small precision (small correctness of generated questions).

On the other hand, lexico-syntactic rules enriched by information about entities help to generate good questions, but the coverage of input sentences is weak. So, it is a tradeoff between correctness of question and coverage. And with increasing number of patterns, another problem arises: how to store and search among large number of patterns. Usage of machine learning methods for patterns obtaining was recently tested in IBM Watson [11]. They used a dataset consisting of thirty million pairs of RDF triplets and questions created by [15]. But questions generating from text is a little more complicated task in comparison to the generating question from structured (RDF) data.

Based on these problems, we defined main goal of our research as follows:

- Design of a question generation method which will use combination of linguistics approaches and machine learning methods to eliminate drawbacks of systems based purely on hand-written transformation rules.

We are expecting, that combination of these approaches could eliminate the drawbacks when the approaches are used separately. Database of transformation rules could be built automatically by initial sentence-question pairs (known as supervised learning) and continuous enhancement of the model could be based on feedback about already generated questions (known as reinforcement learning). We also set several requirements for our method:

- input is an unstructured text (e.g. sentence or paragraph),
- output is a question in a sentence form (e.g. interrogative sentence),
- learning and improvement of transformation rules will be obtained automatically by sentence-question pairs without the intervention of a human expert (without modification of source code or manual creation of transformation patterns),
- question filtering and question selection will be based on learned language and statistical models.

Whereas the main goal is relatively robust, we decided to split it into several scientific and implementation goals.

2.1 Scientific Goals

- Detailed analysis of the state-of-the-art question generation systems and identification of its drawbacks.
- Train the model for AQG task based on data-driven approach.
- Realization of experiments which confirms the ability of proposed method to learn how to create questions.
- Realization of experiments with purpose to compare questions generated by our method with questions generated by the state-of-the-art systems.
- Realization of experiments aiming to compare question generated by AQG systems with questions expected by users.

2.2 Implementation Goals (AQG Framework)

- Create question generation framework (described in our previous work [2]).
- Build (find or create) a dataset for learning the framework, how to create questions from text.
- Create online interface for evaluating the questions.

We demonstrate the fulfillment of our intended goals in the following sections.

3. Our Proposed AQG Method

In this section, we will briefly introduce our proposed question generation method and its implementation in our question generation framework. The framework was build not just for training and testing of our method, but also for evaluations of generated questions and several user experiments described later. It consists of three modules:

- preprocessing module which transforms input text into special data structure representing input sentences called “composite pattern” (marked as CP),
- question generation module which creates the questions based on CP,
- question estimator which estimates the quality of generated questions, remove duplicate and incorrect questions and sort the questions by estimated quality.

Table 1: Composite Pattern (CP) as a Special Data Structure for Automatic Question Generation Task

Sentence	Miroslav	was	a	student	in	Bratislava,	Slovakia
Lemma	Miroslav	be	a	student	in	Bratislava,	Slovakia
POS	NNP	VBD	DT	NN	IN	NNP	NNP
POS simplified	NN	VB	DT	NN	IN	NN	NN
NER	Person	-	-	-	-	Location	Location
Wordnet	-	-	-	-	-	Location	Location
GKG	-	-	-	-	-	City	Country
...	...	...	...	...	...	...	...

3.1 Preprocesing Module

This module is responsible for extracting the information from input sentences and for the creation of CP. CP is a matrix of information about each token in the sentence and its structure is shown in Table 1. The rows of the table represent patterns of the sentence and columns are labels of each token of pattern. Preprocessing task consists of several steps: splitting text into sentences, splitting sentence into tokens, identify information for each token (part-of-speech labels (POS), named entities (NER), semantic information from WordNet and some related databases (e.g. Viaf, Google Knowledge Graph etc.) and identifying dependency relationships between tokens. Token dependencies are used for simplification of complex sentences (sentences with appositions or multiple sentences joined with conjunctions).

CP represents the structure of sentence and it is used for searching suitable transformation rules in the following question generation phase. There are hierarchical relationships between patterns in CP which can be represented in tree structure (figure 1)[4]. The pattern on the top (POS pattern) is the same for several detailed patterns which can be different on NER attributes. For example, these two sentences have the same POS pattern (first line in hierarchy on the figure 1), but there are differences in NER pattern (the last token is Person in the first sentence and Location in the second one):

- The first president of Slovakia was Rudolf Schuster.
- The capital city of Slovakia is Bratislava.

The output of this module is a sequence of sentences represented in CP and these data structures are then used for sentence-question transformations. Contributions of various levels of patterns stored in CP and matching process of sentences is described in our previous work [4].

3.2 Question Generation Module

Transformation rules for question generation are learned during training phase by initial pairs of sentences and questions assigned to these sentences. From the both parts of data (input sentences and questions), the CP are obtained, and model extract and stores the list of transformation steps based on the difference between these CP. It just stores standard set of actions known from similarity matching algorithms (insert token, remove token, replace token or change the position of token). Later, we also added one special operation for transforming the form of token (e.g. change the singular form into plural form or change the tense of verb). Output of training phase is a model which stores list of transformation actions between sentences and questions and these actions are used for sentence to question transformation. Transformation patterns for new sentences are chosen by similarity between these sentences and already stored sentences. This process is based on mathematical calculation of difference between their CP (similarity is calculated as a portion of identical tokens).

3.3 Filtering Module

Filtering module estimates the correctness of generated questions. It is based on statistical information obtained from the past (how many similar questions were created and what correctness value they obtained). Then it filters away the duplicate questions and sort out the remaining questions by estimated score. Calculation of score also reflect similarity between current CP and CP of questions generated in the past. Maximum value of score is one (which means, that system generated question with the same CP in the past). We have made several experiments with focus to determine the ideal threshold, but it is still tradeoff between correctness of generated questions and coverage of various input sentences. When the score was set too high, the system generated more correct question, but the total number of questions was lower.

4. Experiments and Evaluations

We have made several experiments to verify our question generation approach. Firstly, we have focused on our method and confirmation of our hypothesis saying that it is possible to train the model which will be able to generate questions from unstructured text. We have also evaluated filtering module and determined that the ideal threshold value for filtering the questions is 0.6 (questions with lower matching score are usually incorrect). Then we let the users compare questions generated by our method and question generated by the best state-of-the-art systems [14]. These results were used for comparisons of question generation systems and for the comparison of the questions generated by AQG systems with questions expected by users.

4.1 Evaluation of the Proposed AQG Method

The first experiment was done with the purpose to find out, if it is possible to train the model for AQG task. For this experiment we used large dataset of sentence-question pairs like in [15] (which was successfully used for training question generation model from structured data), but with pairs of sentences and questions. Dataset used in QGSTEC [13] is too small (it contains only 180 sentences and questions in total), so we decided to let this dataset for system comparisons and we built new dataset for training phase. Our created dataset consists of 206 articles about countries in the world (articles from Wikipedia). The first quarter of articles were used for training and the rest for testing. We created an interface where the sen-

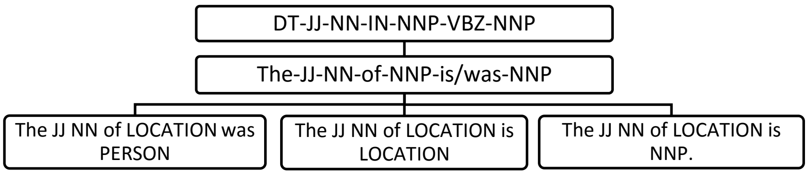
DT-JJ-NN-IN-NNP-VBZ-NNP
The-JJ-NN-of-NNP-is/was-NNP
The JJ NN of LOCATION was PERSON
The JJ NN of LOCATION is LOCATION
The JJ NN of LOCATION is NNP.

Figure 1: Hierarchy of sentence patterns based on level of abstraction [4]**.**

Table 2: Briefly Comparison of AQG Systems: Number of Generated Questions and Question Correctness

	questions	syntactic correct.	semantic correct.
A	320	0.21	0.17
B	165	0.49	0.45
M	100	0.74	0.68

tences from articles were shown and the users could write their questions. Our framework created CP for the input sentences and also fo the questions obtained by users and then stored the transformation pattern. Then we imported sentences from the testing dataset and let the system generate questions on transformation rules learned during training phase. Our results confirmed, that the system could generate questions for new (unseen) sentences.

The overall quality of generated questions was evaluated by another group of users and the average correctness of question was 87,9 %. We have published detailed information about dataset creation, training the model, determining the quality threshold and evaluating of question generation in our previous paper [2].

4.2 Comparison of Question Generation Systems

The second experiment was done with the purpose to compare quality of questions generated by our method with questions generated by the best state-of-the-art systems [14]. Whereas scoring criteria used on QGSTEC event was complicated and hardly replicable, we decided to import all questions to our interface and let the user evaluate all questions. The evaluation of the questions consisted of two criteria: syntactic correctness (correct grammar of question) and semantic correctness (intelligibility of question). Evaluators could rate on the three-level scale for each attribute: correct (+1), incorrect (-1), almost correct (+0.5) and they also could 'do not answer' (+0) if they were no sure about correctness.

In table 2, we present results of this experiment (detailed results are discussed in our previous paper [3]). As we can see, our AQG system generates more correct questions, but on the other hand, the total number of generated questions was lower. Globally, our questions obtained the smallest number of bad score and the highest number of good score (Figure 2).

4.3 Comparison of Question Generated by Systems with Questions Expected by Users

The last experiment, which is directly connected to our goal, was done with the intent to compare questions generated by AQG systems with questions expected by users. There were many generated questions which were evaluated as correct (because they were correct), but their form was very untypical. For this purpose, we adopt the metrics used in related NLP tasks (e.g. machine translation or text summarization) to calculate the difference between text generated by human and text generated by machines (BLEU and ROUGE). We found out that there were no significant differences between systems (when we compare only the best questions). Difference between generated and expected questions was approximately 75 percent (BLEU) and 83 percent (ROUGE).

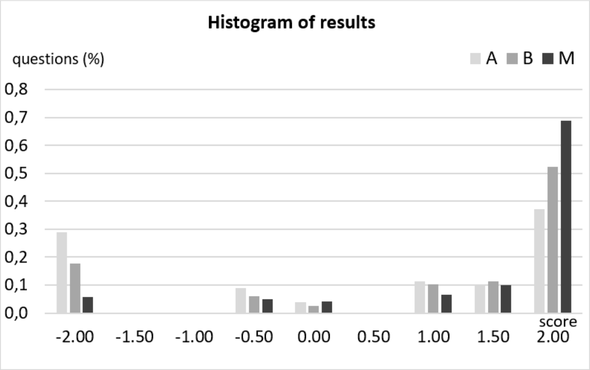

Figure 2: Histogram of votes obtained by user evaluators for each AQG system: our system (M) obtained more positive points and less negative points in comparison to other state-of-the-art systems.

5. Conclusions, Contributions and Future Challenges

In this extended abstract, we have presented our contribution to the AQG task as a task of NLP. Although AQG is a multidisciplinary task based on several research areas (informatics, education, linguistics and psychology), we focused on this task mainly from the NLP point of view. We proposed our novel question generation approach for factual question generation from unstructured text and we also created question generation framework which was used for experiments and evaluations of this method. Other contributions of our work are datasets of sentence-questions pairs which can be used for training new models or for evaluation AQG systems.

Our main research goal was to create a method combining traditional approach based on sentence patterns with data-driven approaches known from machine learning. It will allow us to train and improve method by sentence-question pairs instead of manual creation of transformation rules. We also proposed special data structure called composite pattern, which keeps set of several sentence patterns and it helps to match the new (unseen) sentences with learned transformation rules. Our question generation framework uses question estimator module. Based on its estimated score of correctness, we can filter out many duplicate or incorrect questions and increase the portion of correct questions.

We have made several experiments which confirm that our model is able to learn how to create questions from sentences and question generated by our method achieve better correctness in comparison to the best of state-of-the-art systems from the Question Generation Shared Task Challenge. We also compare questions generated by AQG systems with questions expected by users through metrics used in related NLP tasks and found, that there are only small differences between them.

There are many challenges in AQG area. The biggest limitation of this task is a lack of training data and shortcomings of external NLP tools, which are essential for this task. The quality of generated questions depends on annotation of input text and the correct identification of sentence structure (e.g. part-of-speech identification, named entity recognition, word dependencies etc.). This also applies to the extensibility of our approach for other languages. In our work, we focus on English language. But it will be interesting to train the model on sentence-questions pairs from different language. Unfortunately, our effort to train the model for Slovak language failed due to the NLP tools, whereas it is very difficult to extract structure of Slovak sentence (extract information about entities, part-of-speech tags etc.). On the other hand, our approach is language-independent, so if there are available tools in the future, it would be interesting to do some experiments.

Thanks to the advancement of research in the field of automatic question generation, the software systems will be able to provide a great support of knowledge verification.

Acknowledgements. The work reported here was supported by the Scientific Grant Agency of Slovak Republic (VEGA) under the grant number VG 1/0646/15, ITMS 26240120039, Slovak Research and Development Agency under the contract No. APVV-16-0213, Cultural and Educational Grant Agency of the Slovak Republic (KEGA), grant No. 019STU-4/2018 and STU Grant scheme for Support of Young Researchers.

References

[1] M. Baylor. Usability study on a website integrating self organized learning environments (sole) and google apps for education (gafe). 2017.

[2] M. Blšták and V. Rozinajová. Machine learning approach to the process of question generation. In *International Conference on Text, Speech, and Dialogue*, pages 102–110. Springer International Publishing, August 2017.

[3] M. Blšták and V. Rozinajová. Building an agent for factual question generation task. In *World Symposium on Digital Intelligence for Systems and Machines, DISA 2018*. [accepted], 2018.

[4] M. Blšták and V. Rozinajová. Automatic question generation based on analysis of sentence structure. In *International Conference on Text, Speech, and Dialogue*, pages 223–230. Springer International Publishing, September 2016.

[5] Y. Chali and S. A. Hasan. Towards topic-to-question generation. *Computational Linguistics*, 41(1):1–20, 2015.

[6] S. Curto, A. C. Mendes, and L. Coheur. Question generation based on lexico-syntactic patterns learned from the web. *Dialogue & Discourse*, 3(2):147–175, 2017.

[7] M. Heilman and N. A. Smith. Question generation via overgenerating transformations and ranking. Technical report, Carnegie-Mellon University of Pittsburgh, 2009.

[8] M. Heilman and N. A. Smith. Extracting simplified statements for factual question generation. In *Proceedings of QG2010: The Third Workshop on Ques-tion Generation*, volume 11, 2010.

[9] Y. T. Huang, Y. M. Tseng, Y. S. Sun, and M. C. Chen. Tedquiz: Automatic quiz generation for ted talks video clips to assess listening comprehension. In *2014 IEEE 14th International Conference on Advanced Learning Technologies*, pages 350–354, July 2014.

[10] M. Khvalchik and A. Kulkarni. Open-domain non-factoid question answering. In *International Conference on Text, Speech, and Dialogue*, pages 290–298. Springer, 2017.

[11] J. Lee, G. Kim, J. Yoo, C. Jung, M. Kim, and S. Yoon. Training ibm watson using automatically generated question-answer pairs. *arXiv preprint arXiv:1611.03932*, 2016.

[12] S. Reddy, D. Raghu, M. M. Khapra, and S. Joshi. Generating natural language question-answer pairs from a knowledge graph using a rnn based question generation model. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, volume 1, pages 376–385, 2017.

[13] V. Rus and C. G. Arthur. The question generation shared task and evaluation challenge. In *The University of Memphis. National Science Foundation*. Citeseer, 2009.

[14] V. Rus, B. Wyse, P. Piwek, M. Lintean, S. Stoyanchev, and C. Moldovan. A detailed account of the first question generation shared task evaluation challenge. *Dialogue & Discourse*, 3(2):177–204, 2012.

[15] I. V. Serban, A. García-Durán, C. Gulcehre, S. Ahn, S. Chandar, A. Courville, and Y. Bengio. Generating factoid questions with recurrent neural networks: The 30m factoid question-answer corpus. *arXiv preprint arXiv:1603.06807*, 2016.

[16] E. Sneiders. Automated faq answering with question-specific knowledge representation for web self-service. In *Human System Interactions, 2009. HSI'09. 2nd Conference on*, pages 298–305. IEEE, 2009.

Selected Papers by the Author

M. Blšták and V. Rozinajová. Machine Learning Approach to the Process of Question Generation. *International Conference on Text, Speech, and Dialogue*, pages 102–110, Prague, Czechia, 2017. Springer.

M. Blšták and V. Rozinajová. Automatic question generation based on analysis of sentence structure. In P. Sojka, A. Horák, I. Kopecek, K. Pala, editors, *International Conference on Text, Speech, and Dialogue*, pages 223–230, Brno, Czechia, 2016. Springer.

M. Blšták and V. Rozinajová. Využívanie hierarchie vetných vzorov pre automatizovanú tvorbu otázok. M. Bieliková, I. Srba, editors, *Proceedings WIKT and DaZ 2016. 11th workshop on intelligent and knowledge oriented technologies, 35th conference on data and knowledge*, pages 235–240, Smolenice, Slovakia, 2016.

M. Blšták and V. Rozinajová. Automatizovaná tvorba otázok z textu analýzou štruktúry viet. *Proceedings WIKT 2015: 10th workshop on intelligent and knowledge oriented technologies*, pages 49–52, Košice, Slovakia, 2016.

M. Blšták and V. Rozinajová. Building an Agent for Factual Question Generation Task. *World Symposium on Digital Intelligence for Systems and Machines, DISA 2018*, Košice, Slovakia, 2018. [accepted].

M. Blšták and V. Rozinajová. Automatic Question Generation Based on Sentence Structure Analysis using Machine Learning. *Natural Language Engineering*. [in review].