ICIC Express Letters
Volume **14**, Number **11**, November **2020**

ICIC International ©2020 ISSN 1881-803X
pp. 1059–1067

AUTOMATED QUESTION GENERATING METHOD BASED ON DERIVED KEYPHRASE STRUCTURES FROM BLOOM'S TAXONOMY

Bambang Dwi Wijanarko[1,2], Yaya Heryadi[1], Hapnes Toba[1,3]
and Widodo Budiharto[1]

[1]Computer Science Department, BINUS Graduate Program – Doctor of Computer Science
[2]Computer Science Department, BINUS Online Learning
Bina Nusantara University
JL. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta 11480, Indonesia
{ bwijanarko; yayaheryadi; wbudiharto }@binus.edu

[3]Faculty of Information Technology
Maranatha Christian University
Jl. Suria Sumantri 65, Bandung 40164, Indonesia
hapnestoba@it.maranatha.edu

Received January 2020; accepted April 2020

Abstract. *Learning systems for higher education have improved significantly in the past decade. A central research topic is how to stimulate learning and improve the quality of questions for learning evaluation by generating relevant questions with objective of learning and cognitive skill levels. This study presents an automated question generating method based on keyphrase to improve learning system in such way that the generated questions comply with both predetermined learning outcomes and Bloom's taxonomy of hierarchical cognitive skill levels. Unlike the prominent methods, this study proposed the learning materials written in local languages transformed into English which make the method applicable to learning material written in various languages given the language translation to English is available. In addition, the expert knowledge about keyphrase patterns in this study is represented in a set of context-free grammar rules making it possible to extend the pattern's rules to capture various keyphrase structures. The study which used 1,432 sentences from learning material written in Bahasa Indonesia can generate about 60,000 questions. From the experimentation, the average of Bilingual Evaluation Understudy (BLEU) score which is computed from the generated questions is 0.9566. This result indicates that the generated questions are well understood by human assessors. From sentiment analysis results, the generated questions achieve 0.73 classified as neutral, and 0.68 classified as objective. The Cohen-Kappa coefficient of 0.34589 means there is a difference between human evaluators in understanding the generated questions.*
Keywords: Question generating, Keyphrase extraction, Keyphrase template, Bloom's taxonomy, Context-free grammar

1. **Introduction.** Question generating task is an interesting research problem in Natural Language Processing (NLP) research domain. The aim of the question generating is to generate a question or interrogative sentence from some given texts as input. In the past ten years, the question generating task has gained wide attention from researchers in education domain. A study reported by [1] showed question generating has been adopted to support effective teaching mainly to promote critical thinking, retention, and engagement. Among a variety of approaches to question generating, the keyphrase structure-based method is very prospective due to its ability to capture syntax and semantics of the input text. However, key phrase extraction remains a challenging research question. Several attempts to address the keyphrase extraction problem have been proposed. For

DOI: 10.24507/icicel.14.11.1059

example, a method based on an encoder-decoder generative model has been proposed by [2], the model to generate key phrases based on the semantic meaning of the text, and a method which used linguistic tags and rules [3].

Although many prominent studies in generating question for education have been reported to the best of our knowledge, only a few studies proposed question generating methods for learning evaluation of online learning using Bloom's taxonomy [4] to derive the key phrase structure as a pattern for generating relevant questions from textual learning materials. Therefore, the purpose of this study is to propose a question generating method based on a phrase structure template derived from Bloom's taxonomy.

The remaining of this paper is organized as follows. First, some relevant studies in phrase structure-based question generating will be described. Next, research methods will be explained. Finally, the experimentation results will be discussed followed by a conclusion and future works.

2. **Theoretical Foundation.**

2.1. **Related work.**

A plethora of question generating methods can be categorized broadly into three categories namely: 1) factoid questions that ask fact-based answers, 2) list questions that ask a set of answers, and 3) other questions [5]. Based on the generated questions, these methods can be divided further into two categories namely 1) deep question category in which to answer the questions requires more logical thinking [6], for example: why, why not, what if, what if not, and how type of questions; and 2) shallow question category in which the generated questions focus on facts, for example: who, what, when, where, which, how many, how much, and yes/no type of questions.

Among deep question generating methods, the keyphrase structure-based method is a prospective method application in education domain mainly due to its ability to capture syntax and semantics of the input text by exploiting main keyphrase existing in the sentence. Several attempts to address the keyphrase extraction have been proposed. For example, the study by Turney [7] proposed a coherent keyphrases extraction method based on statistical association between candidate phrases to determine their semantic relevance. The study by [2] proposed a feature extraction method using machine learning approach in which keyphrase is predicted using encoder-decoder generative model [8]. The study by [3] proposed feature extraction method using linguistic tags and rule. In the study reported by [9], phrase structures are extracted using essential feature in $TF \times IDF$ document representation matrix. The method which utilizes key phrase embeddings for unique key phrase extracted from scientific articles followed by key phrase ranking using PageRank is proposed by [10]. Finally, the recent study on feature extraction used machine learning to learn end-to-end neural question generating systems to produce pairs of questions and answers through context paragraph input [1]. The generated questions are further evaluated by education experts.

Despite its accuracy, the machine learning approach for question generating in learning evaluation context has several drawbacks mainly the resulted phrase structures have limited question format, syllabus coverage, difficulty level, and cognitive level, according to Bloom's taxonomy [11]. The problem of understanding natural language automatically has long been a challenge for NLP research. Gan and Yu models [12] the understanding of geometry questions as a problem of extracting entity relations through a syntactic-semantic model approach.

2.2. **Keyphrase.**

Keyphrase is a term defined as a "*high-level description of a document's contents*". The term is widely used in several document analyses and modelings such as document summarization, clustering, and topic search. Despite its wide applications, the keyphrase extraction process is a challenging task. The study result reported by Mikolov et al. [13] proposed a phrase extraction method based on probability of the two-word sequence which is measured as follows:

$$score(w_i, w_j) = \frac{count(w_i w_j) - \delta}{count(w_i) \times count(w_j)} \tag{1}$$

where $count(w_i w_j)$ is the number of a sequence of word w_i followed by w_j, $count(w_i)$ is the number of words w_i, $count(w_j)$ is the number of words w_j in the input document, and δ is a constant to prevent too many phrases which are formed by some very less frequent words. Another method proposed by Turney [7] treated the problem as a classification task. In the proposed method, the author proposed Naive Bayes model as the keyphrase classifier. The proposed method comprises three steps.

1) Document preprocessing such as: removing stop words, removing non-alphanumerics, splitting text into phrase, and phrase stemming.
2) Extracting $TF \times IDF$ score from each phrase sample. The score is a standard metric in information retrieval designed to measure how specific a phrase P in the document D which is formulated as

$$TF \times IDF(P, D) = Pr[phrase\ in\ D\ is\ P] \times -\log Pr[P\ in\ a\ document] \tag{2}$$

where $Pr[phrase\ in\ D\ is\ P]$ is probability counting the number of times the phrase P occurs in the document D, and $Pr[P\ in\ a\ document]$ counting the number of documents in the training corpus that contain P (excluding D).
3) Computing distance of each phrase sample from the beginning of the corresponding document. The distance is computed as the number of words that precede its first appearance, divided by the number of words in the document. A resulting feature is a number between 0 and 1 that represents the proportion of the document preceding the phrase's first appearance.

In this method, the attribute, $TF \times IDF$ and the distance are assumed to be independent because the phrase has been discrete $TF \times IDF$ with value T and distance to values D, then the probability that a phrase is a unique phrase is

$$Pr[key|T, D] = \frac{Pr[T|key] \times Pr[D|key] \times Pr[key]}{Pr[T, D]} \tag{3}$$

where $Pr[T|key]$ is the probability that a keyphrase has a $TF \times IDF$ with score T, $Pr[D|key]$ the probability that it has a distance D. $Pr[T|key]$ is the probability a priori that the phrase is keyphrase, and $Pr[T, D]$ is the normalization factor, so that the value of $Pr[key|T, D]$ is between zero and one.

2.3. **Context-free grammars.** A Context-Free Grammar (CFG) in formal language [14] is defined as a quadruple $G = (V, \Sigma, P, S)$ where V is a set of non-terminal symbols; Σ is a set of terminals where $V \bigcap \Sigma = \emptyset$; P is a set of rules P: $V \to (V \bigcup \Sigma)^*$, i.e., the left-hand side of the production rule P does have any right context or left context; and S is the start symbol.

In a formal language, CFG rules can be viewed as a set of recursive writing rules that is used to generate strings based on patterns in the set of rules. Hence, CFG is a base for many programming language syntaxes which provide efficient and straightforward parsing algorithms for programming languages, files, and data streams [15]. In this study, CFG is used to develop question templates based on a phrase structure and operational verb in Bloom's taxonomy.

3. Methodology.

3.1. **Research framework.** The question-generating framework in this study can be described in the framework (see Figure 1) which consists of the following main processes: 1) Text Document Translation ID, EN: to translate input textual documents written in Bahasa Indonesia to English, 2) Phrase Construction based on Phrase Pattern: to develop phrase structure grammar as template/pattern, 3) Template-based Question Generating: to generate question candidates based on list of phrase and Bloom's taxonomy, 4) Text Document Translation EN, ID: to translate the generated questions written in English to Bahasa Indonesia, and 5) Question Evaluation: to evaluate the generated question using BLEU score.

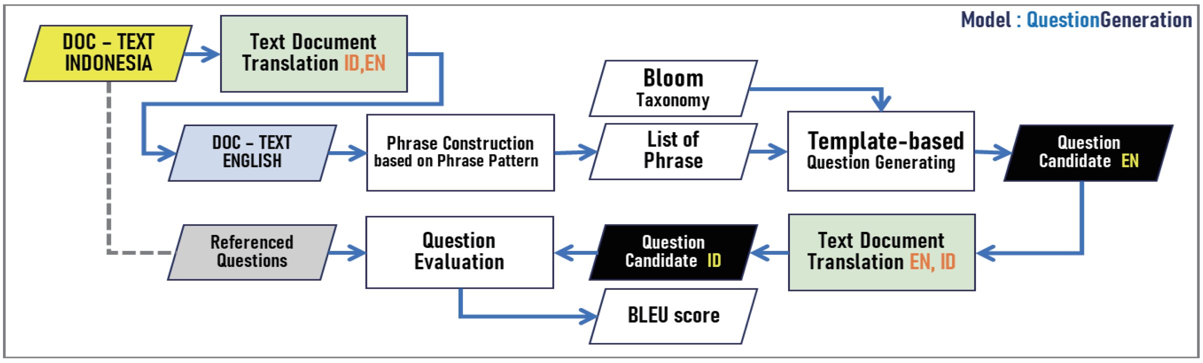

Model : QuestionGeneration
DOC – TEXT INDONESIA
Text Document Translation ID,EN
Bloom Taxonomy
DOC – TEXT ENGLISH
Phrase Construction based on Phrase Pattern
List of Phrase
Template-based Question Generating
Question Candidate EN
Referenced Questions
Question Evaluation
Question Candidate ID
Text Document Translation EN, ID
BLEU score

FIGURE 1. Automatic question generating based on keyphrase structure model

3.2. **Dataset.** Following [4], this study uses a dataset from the learning management systems repository of Binus Online Learning. The dataset comprises textual learning materials of software engineering course. The dataset preprocessing is mainly: 1) removing all images and special characters from the text, and 2) translating the dataset into English so that the text processing can exploit NLTK and TextBlob libraries.

3.3. **Keyphrase extraction.** This step aims to extract keyphrases from input textual documents. In general, it is called keyphrase extraction in this study. Firstly, the input document dataset in Bahasa Indonesia is translated into English. The results are splitted into 10 main learning topics. The splitted documents are then processed using TextBlob Python library, which provides functionalities such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translations. The noun phrases are extracted from the document samples using chunk parsing of TextBlob library that has been trained with the Corpus ConLL-2000. This process sequences generated 3,345 unique keyphrases consisting of two or more words.

3.4. **Template construction and question generating.** This step aims to construct a keyphrase template based on the English phrase structure and Bloom's taxonomy and generating questions based on the keyphrase templates. This step comprises two main steps. *First*, setting out context-free grammar rules to represent main phrase structure of English language chosen for this study. In these rules, NNP, NNI, JJ, and VB post-tag variables are chosen as the main variables that generated strings. The main context-free grammar rules to generate keyphrase structure are as follows.

$$\text{NNP} \rightarrow \text{NNP NNP} \tag{4}$$

$$\text{NNI} \rightarrow \text{NN} \mid \text{NNI NN} \mid \text{NN NNI} \mid \text{JJ NN} \mid \text{NN JJ} \tag{5}$$

$$\text{JJ} \rightarrow \text{JJ JJ} \tag{6}$$

$$\text{VB} \rightarrow \text{NN VB} \mid \text{NN VBG} \mid \text{NN VBZ} \mid \text{NN VBP} \mid \text{NN VBN} \mid \text{NN VBD} \tag{7}$$

where NNP is proper noun, singular, NN is noun, singular or mass, JJ is adjective, and VB is verb.

Second, generating keyphrases structure based on Bloom's taxonomy and generating questions based on the keyphase strcutres. Any keyphrase that already has a post-tag component {NNP, NNI, JJ, VB} is used to build templates with combination with operational verbs on Bloom's taxonomy along with additional question words (Table 2) thus generating a keyphrase-based question template model, as below (Table 1).

Table 1. Phrase based on CFG

Adjective Phrase	Noun Phrase
PK010 = {'JJ', 'NN', 'NNS', 'VBG'}	PK041 = {'NN', 'VB'}
PK011 = {'JJ', 'NN', 'NNS'}	PK050 = {'NN', 'VBG'}
PK012 = {'JJ', 'CD'}	PK060 = {'NN', 'VBZ'}
PK020 = {'JJ', 'NN'}	PK070 = {'NN', 'VBP'}
PK030 = {'JJ', 'NNS'}	PK080 = {'NN', 'VBN'}
	PK090 = {'NN', 'VBD'}
	PK091 = {'RB', 'VBN'}
	PK100 = {'NNS', 'VBP'}
	PK101 = {'NN', 'NN'}
	PK102 = {'NN', 'NN', 'NNS'}
	PK103 = {'NNS', 'VBP'}
	PK104 = {'NNS', 'VBZ'}

Table 2. Question template based on phrase structure

No	**Q(X): Question template based on phrase structure**	**Level taxonomy**
1	Describe what is X	1
2	What are the types X	2
3	Mention some example X	2
4	Explain and give examples of X	2
5	Explain the purpose of X	3
6	Why X	4
7	What happens in X	3
8	What is the effect of X	4
9	Why there is an X	4
10	How to X	3
11	What is the importance of X	2
12	How to find out X	4

3.5. **Question generating evaluation.** This step aims to evaluate text fragments for positive and negative subjective expressions and their strength in automatic question-generation. Three metrics are computed to measure the performance of the question generating method as follows. *First*, sentiment analysis aims to find out the polarity and subjectivity of the question. Whilst polarity measures positive, neutral, and negative polarity of the generated questions; subjective measurement measures acceptability of the generated questions. Polarity and subjectivity information can use as a basis for decision making in selecting questions that are considered relevant to the context of the learning material [16]. Polarity and subjectivity are also used to figure out the distribution of sentiment on each level of Bloom's taxonomy. *Second*, BLEU and Cohen's Kappa scores are used to measure the level of reading comprehension questions [17], with the following formula:

$$p_n = \frac{\sum_{C\in\{Candidates\}} \sum_{Ngram\in C} Count_{clip}(Ngram)}{\sum_{C'\in\{Candidates\}} \sum_{Ngram'\in C'} Count_{clip}\,(Ngram')} \tag{8}$$

where $Count_{clip}(Ngram)$ is the maximum number of $Ngram$ occurring in each candidate question, and $Count_{clip}\,(Ngram')$ is the maximum number of $Ngram'$ occurring in each reference question.

$$BP = \begin{cases} 1, & c > r \\ e^{(1-r/c)}, & c \leq r \end{cases} \tag{9}$$

$$BLEU = BP \times e^{\left(\sum_{n=1}^{N} w_n \log p_n\right)} \tag{10}$$

where BP (brevity_penalty) is the length of the reference which is nearest/sufficient, c is the length of the candidate question, r is the length of the reference question, w_n is a weight which is typically computed as $1/N$, and N is the number of questions.

Third, Cohen's Kappa between two challengers whom each classifies the items N into mutually exclusive C categories can be formulated using the following formula:

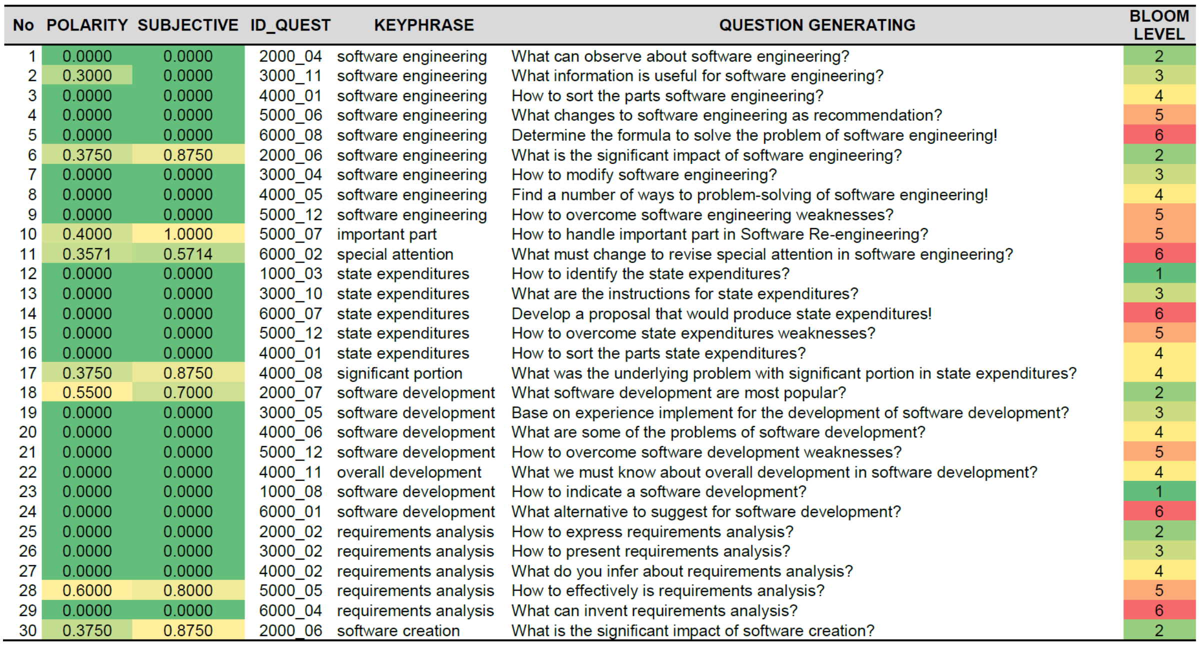
$$\mathcal{K} = \frac{p_o - p_e}{1 - p_e} \tag{11}$$

where p_o is the relative observed agreement among raters (identical to accuracy), and p_e is the theoretical probability of chance agreement.

4. **Result and Discussion.** The dataset as input for this study comprises learning materials of Binus Online Learning in Software Engineering course written in Bahasa Indonesia. The total raw textual documents contain 1,432 sentences that are fed into the data processing sequences. Finally, the keyphrase-based automatic question-generation generates 64,259 questions that correspond to the level of Bloom's taxonomy (see Table 3). As shown in Table 4, Bloom's level distribution of the generated question set is spread proportionally with the highest proportions being Bloom's level 4, 3, and 5.

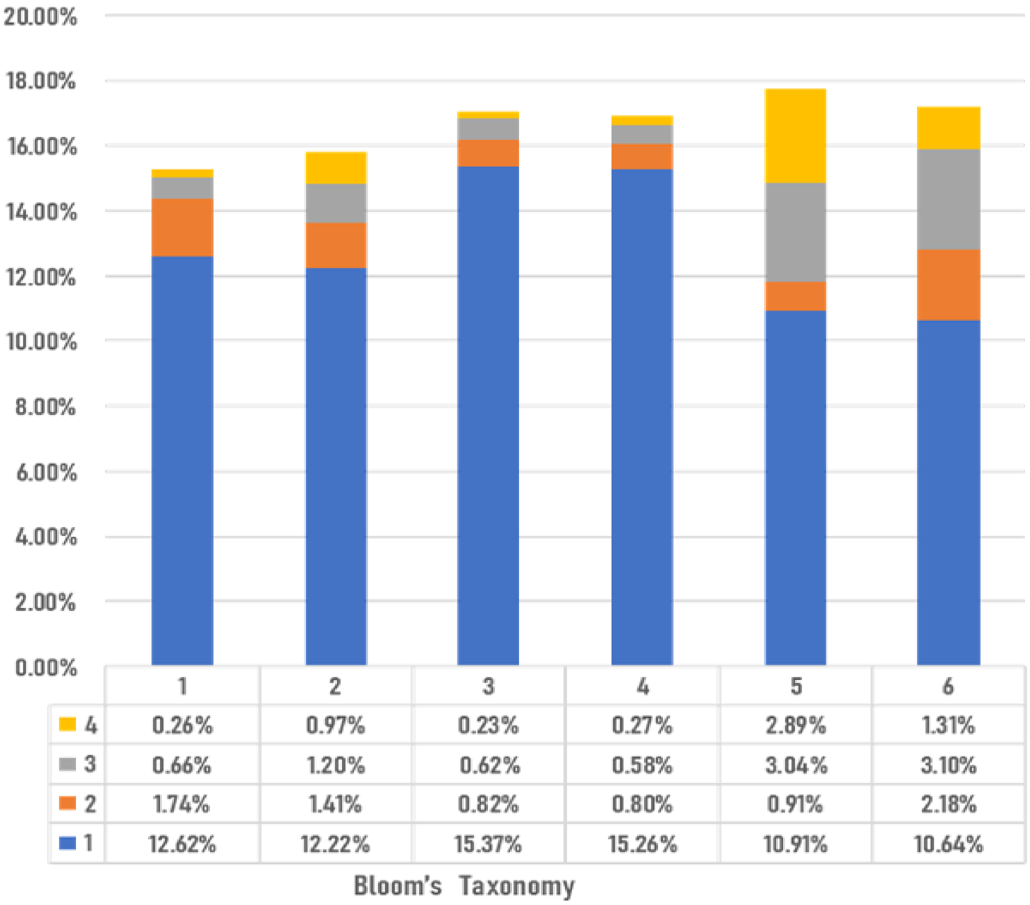
The results of the sentiment analysis toward the generated question set showed that the objectivity score is 77.01%, and neutral polarity of 80.04% which means that the generated questions can be generally accepted (see Table 5). As can be seen from Table

Table 3. Question output from AQG system



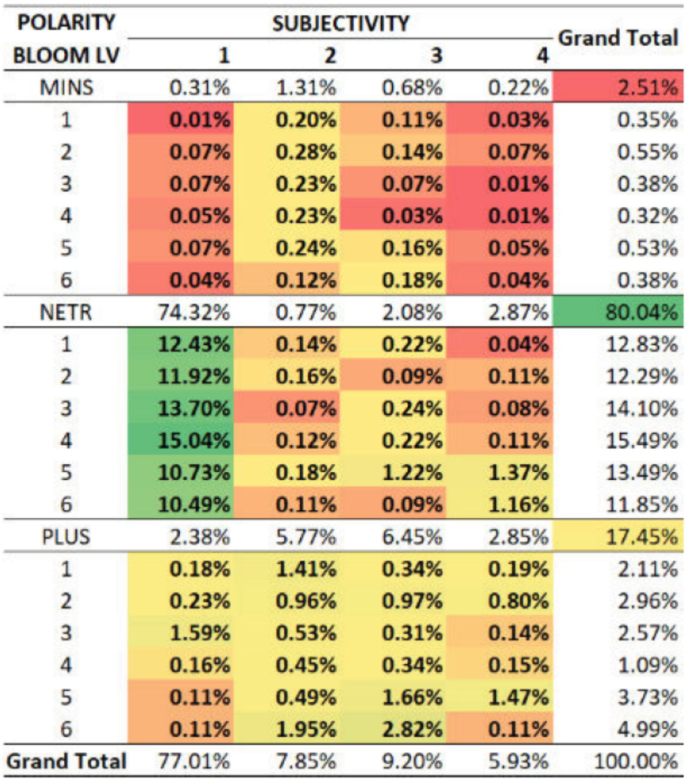
No	POLARITY	SUBJECTIVE	ID_QUEST	KEYPHRASE	QUESTION GENERATING	BLOOM LEVEL
1	0.0000	0.0000	2000_04	software engineering	What can observe about software engineering?	2
2	0.3000	0.0000	3000_11	software engineering	What information is useful for software engineering?	3
3	0.0000	0.0000	4000_01	software engineering	How to sort the parts software engineering?	4
4	0.0000	0.0000	5000_06	software engineering	What changes to software engineering as recommendation?	5
5	0.0000	0.0000	6000_08	software engineering	Determine the formula to solve the problem of software engineering!	6
6	0.3750	0.8750	2000_06	software engineering	What is the significant impact of software engineering?	2
7	0.0000	0.0000	3000_04	software engineering	How to modify software engineering?	3
8	0.0000	0.0000	4000_05	software engineering	Find a number of ways to problem-solving of software engineering!	4
9	0.0000	0.0000	5000_12	software engineering	How to overcome software engineering weaknesses?	5
10	0.4000	1.0000	5000_07	important part	How to handle important part in Software Re-engineering?	5
11	0.3571	0.5714	6000_02	special attention	What must change to revise special attention in software engineering?	6
12	0.0000	0.0000	1000_03	state expenditures	How to identify the state expenditures?	1
13	0.0000	0.0000	3000_10	state expenditures	What are the instructions for state expenditures?	3
14	0.0000	0.0000	6000_07	state expenditures	Develop a proposal that would produce state expenditures!	6
15	0.0000	0.0000	5000_12	state expenditures	How to overcome state expenditures weaknesses?	5
16	0.0000	0.0000	4000_01	state expenditures	How to sort the parts state expenditures?	4
17	0.3750	0.8750	4000_08	significant portion	What was the underlying problem with significant portion in state expenditures?	4
18	0.5500	0.7000	2000_07	software development	What software development are most popular?	2
19	0.0000	0.0000	3000_05	software development	Base on experience implement for the development of software development?	3
20	0.0000	0.0000	4000_06	software development	What are some of the problems of software development?	4
21	0.0000	0.0000	5000_12	software development	How to overcome software development weaknesses?	5
22	0.0000	0.0000	4000_11	overall development	What we must know about overall development in software development?	4
23	0.0000	0.0000	1000_08	software development	How to indicate a software development?	1
24	0.0000	0.0000	6000_01	software development	What alternative to suggest for software development?	6
25	0.0000	0.0000	2000_02	requirements analysis	How to express requirements analysis?	2
26	0.0000	0.0000	3000_02	requirements analysis	How to present requirements analysis?	3
27	0.0000	0.0000	4000_02	requirements analysis	What do you infer about requirements analysis?	4
28	0.6000	0.8000	5000_05	requirements analysis	How to effectively is requirements analysis?	5
29	0.0000	0.0000	6000_04	requirements analysis	What can invent requirements analysis?	6
30	0.3750	0.8750	2000_06	software creation	What is the significant impact of software creation?	2

Table 4. Distribution of Bloom's taxonomy



	1	2	3	4	5	6
🟨 4	0.26%	0.97%	0.23%	0.27%	2.89%	1.31%
⬜ 3	0.66%	1.20%	0.62%	0.58%	3.04%	3.10%
🟧 2	1.74%	1.41%	0.82%	0.80%	0.91%	2.18%
🟦 1	12.62%	12.22%	15.37%	15.26%	10.91%	10.64%

Table 5. Distribution of sentiment analysis



POLARITY	SUBJECTIVITY				Grand Total
BLOOM LV	1	2	3	4	
MINS	0.31%	1.31%	0.68%	0.22%	2.51%
1	0.01%	0.20%	0.11%	0.03%	0.35%
2	0.07%	0.28%	0.14%	0.07%	0.55%
3	0.07%	0.23%	0.07%	0.01%	0.38%
4	0.05%	0.23%	0.03%	0.01%	0.32%
5	0.07%	0.24%	0.16%	0.05%	0.53%
6	0.04%	0.12%	0.18%	0.04%	0.38%
NETR	74.32%	0.77%	2.08%	2.87%	80.04%
1	12.43%	0.14%	0.22%	0.04%	12.83%
2	11.92%	0.16%	0.09%	0.11%	12.29%
3	13.70%	0.07%	0.24%	0.08%	14.10%
4	15.04%	0.12%	0.22%	0.11%	15.49%
5	10.73%	0.18%	1.22%	1.37%	13.49%
6	10.49%	0.11%	0.09%	1.16%	11.85%
PLUS	2.38%	5.77%	6.45%	2.85%	17.45%
1	0.18%	1.41%	0.34%	0.19%	2.11%
2	0.23%	0.96%	0.97%	0.80%	2.96%
3	1.59%	0.53%	0.31%	0.14%	2.57%
4	0.16%	0.45%	0.34%	0.15%	1.09%
5	0.11%	0.49%	1.66%	1.47%	3.73%
6	0.11%	1.95%	2.82%	0.11%	4.99%
Grand Total	77.01%	7.85%	9.20%	5.93%	100.00%

3, the highest polarity score of the generated question is neutral followed by positive and negative. The classification results based on sentiment showed that the generated questions belong to more objective groups than subjective ones.

The evaluation results using BLUE scores by five human assessors (human expert) mostly > 0.90 which showed that the expert understanding toward the generated questions is relatively high (see Table 6), and the Cohen's Kappa is 0.35.

Table 6. BLEU score

N-GRAM	BLEU_1	BLEU_2	BLEU_3	BLEU_4	BLEU_5
1	0.9985	0.9505	0.9677	0.9613	0.9613
2	0.9974	0.9389	0.9577	0.9521	0.9521
3	0.9962	0.9273	0.9503	0.9450	0.9450
4	0.9952	0.9144	0.9441	0.9384	0.9384

Cohen's Kappa: 0.34589

Finally, the running test in the laboratory, average processing to generate the question dataset required less than 3 minutes using a laptop-sized with processor Intel-i7 and 16 GB memory.

5. **Implications & Limitations.** The proposed method can generate more than 60,000 open questions automatically from 1,432 input sentences. These generated questions distributed over Bloom's taxonomy levels. Interestingly, the majority of generated questions distributed mostly in levels 3, 4, and 5 of Bloom's taxonomy. These results indicated that the generated questions complied with online learning that emphasizes skills and reasoning power based on learning experiences in the community. With such outputs, the proposed question generating method can improve learning operations effectivity by reducing question generating time and always keep the problem banks updated with the learning materials. Having embedded into a learning management system, the question generating method can leverage learning (e.g., triggering forum discussion) as well as an evaluation process. The limitations of this study are mainly: 1) the keyphrase patterns do not include multiple-choice question types. For this purpose, the keyphrase templates need to be improved to capture more template patterns; and 2) the keyphrase patterns do not include question that requires an illustration such as drawings, tables, or mathematical formulas.

6. **Conclusions.** Question generating has become very instrumental to stimulate the learning process in the education context. In such a process, the use of Bloom's taxonomy in question generating provides learners with various questions that comply with syllabus coverages, difficulty levels, and cognitive levels. In general, this study showed that question generating with transformed learning material from Bahasa Indonesia into English and transforming the generated question into Bahasa Indonesia does not reduce objectivity, polarity, understandability, and acceptability of the generated questions. From learning evaluation coverage, the study results showed that the generated questions covered all learning topics in the learning materials. In addition to these findings, this study also showed that context-free grammar is very effective to represent patterns of the rules for keyphrase extraction from textual input.

The quantitative key findings from this study are, using learning material from Binus Online repository as input dataset, evaluation results toward the proposed method as follows: 1) sentiment analysis achieved 0.68 objectivity score and 0.73 neutral polarity score which means that the generated questions are generally accepted; 2) the average BLEU scores above 0.9566, meaning that automatic question-generation is acceptable and implemented; and 3) Cohen's Kappa coefficient of 0.35. Differences opinions of questions sentence understanding can cross with recommendations or choice of questions in evaluating learning according to Bloom's taxonomy. The qualitative finding is mainly the generated questions fairly distributed tens of thousands of questions over six Bloom's taxonomic levels and represents the context of questions according to the course syllabus. With such findings, the next step of this research is to explore the machine learning approach to the question-generation.

Acknowledgment. This work is supported by Directorate General of Research and Development Strengthening, Indonesian Ministry of Research, Technology, and Higher Education, as a part of Doctoral Dissertation Research Grant to Binus University titled "Automatic Question Generation in Online Learning Using Encoder-Decoder with Attention Mechanism" with contract number: 049/LL3/PG/2020.

REFERENCES

[1] A. Willis, G. Davis, S. Ruan, L. Manoharan, J. Landay and E. Brunskill, Key phrase extraction for generating educational question-answer pairs, *Proc. of the 6th 2019 ACM Conference on Learning at Scale (L@S 2019)*, 2019.

[2] R. Meng, S. Zhao, S. Han, D. He, P. Brusilovsky and Y. Chi, Deep keyphrase generation, *CoRR*, pp.582-592, 2017.

[3] Z. Yang, J. Hu, R. Salakhutdinov and W. W. Cohen, Semi-supervised QA with generative domain-adaptive nets, *ACL 2017 – The 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 2017.

[4] B. D. Wijanarko, D. F. Murad, Y. Heryadi, Lukas, H. Toba and W. Budiharto, Questions classification in online discussion towards smart learning management system, *International Conference on Information Management and Technology (ICIMTech)*, pp.251-255, 2018.

[5] E. M. Voorhees and D. M. Tice, Building a question answering test collection, *Proc. of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval – SIGIR '00*, pp.200-207, 2000.

[6] A. C. Graesser, V. Rus, Z. Cai and X. Hu, Question answering and generation, *Appl. Nat. Lang. Process. Identification, Investigation and Resolution*, no.1, pp.1-16, 2009.

[7] P. D. Turney, Coherent keyphrase extraction via web mining, *International Joint Conference on Artificial Intelligence (IJCAI)*, 2003.

[8] P. Pabitha, S. Suganthi and R. Ram, Key phrase extraction using Naive Bayes in question generation system, *Asian J. Inf. Technol.*, vol.15, no.3, pp.372-375, 2016.

[9] I. H. Emu, A. U. Ahmed, M. Islam, S. Al Mamun and A. Uddin, An efficient approach for keyphrase extraction from English document, *Int. J. Intell. Syst. Appl.*, pp.59-66, 2017.

[10] S. Mandal, S. K. Mahata and D. Das, *Preparing Bengali-English Code-Mixed Corpus for Sentiment Analysis of Indian Languages*, 2018.

[11] V. M. Kale and A. W. Kiwelekar, An algorithm for question paper template generation in question paper generation system, *The International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE 2013)*, pp.256-261, 2013.

[12] W. Gan and X. Yu, Automatic understanding and formalization of natural language geometry problems using syntax-semantics models, *International Journal of Innovative Computing, Information and Control*, vol.14, no.1, pp.83-98, 2018.

[13] T. Mikolov et al., *Distributed Representations of Words and Phrases and Their Compositionality*, 2013.

[14] J. E. Hopcroft, R. Motwani and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, 2nd Edition, ACM SIGACT News, 2001.

[15] S. Moraes, A. Godbole and P. Gharpure, Affinity analysis for context-free grammars, *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI 2017)*, pp.2471-2474, 2018.

[16] D. Adamson, D. Bhartiya, B. Gujral, R. Kedia, A. Singh and C. P. Rosé, Automatically generating discussion questions, *AIED*, pp.81-90, 2013.

[17] Y. Gao, L. Bing, W. Chen, M. R. Lyu and I. King, Difficulty controllable generation of reading comprehension questions, *International Joint Conference on Artificial Intelligence (IJCAI)*, pp.4968-4974, 2019.