

Questions Generation for Reading Comprehension using Coherence Relations



Anamika, Vibhakar Pathak, Vishal Shrivastava, Akil Pandey

Abstract: The present scenario requires a large amount of digital content in the context of forums, directories, videos, classes, etc. Questions can thus play a significant role in digital media by quizzes, questions. Producing Question is the job of automatically producing questions from natural language, acting as one of the main fields of natural language human-computer interaction. we focus on generating fact-seeking, questions using the knowledge base. We implemented a system that takes a reading comprehension text as input and outputs all questions for the selected domain. Our system makes the GQ process three-stage.

1. Content selection selected for question generation
2. Question formation (content transformations to get the question),
3. Evaluate the quality of generated question.

The framework system is implemented as an end-to-end system that expects a human to specify a topic. The resulting output is a set of questions in natural language, that Follows the input domain. We show the effectiveness of our approach with previous Heilman and Smith MH method.

Keywords: Natural language generation, question generation, semantic role labeling, templates, self-directed learning

I. INTRODUCTION

1.1 Background Overview

Questions can play a significant role in digital media by quizzes, questions. A considerable amount of research has been invested in the extraction of factual knowledge from unstructured web resources. These efforts resulted in the creation of a knowledge base, which provides this information in a machine-interpretable format. Given this topical diversity, there is great potential for the creation of a system that can facilitate this knowledge for educational purposes. As part of the learning process, the system could generate questions of a certain topic that is adequate to the learner's information need and expertise level. By using automatically generated questions as a medium for knowledge acquisition, a novel utilization for knowledge base could be created.[1]

As stated above, the challenges we address along the way include the generation of the contents of the question, the verbalization of these contents for humans and the judgment of question difficulty. Correspondingly, our contributions fall into the following research areas:

- Question Generation: We propose a novel approach to generate a question, which has a unique answer, using semantic and features based information from the knowledge base.
- Query Verbalization: We elaborate on a pattern-based technique for verbalizing queries, using lexical resources. The resulting natural language mimics the style of clues. To cater to verbalization variety, we expanded the standard set of paraphrases for relations and created a method to distinguish important types for an entity.[2]
- Question Difficulty Estimation: We designed, implemented and evaluated a question difficulty classifier trained on data. The classifier's features are based on statistics computed from the knowledge base.

II. RELATED WORK

We read various papers to Automatic Question Generation listed in the previous section. describes a computer system, Ruminator, which learns by reflecting on the information it has acquired and posing questions in order to derive new information. Ruminator takes as input simplified sentences in order to focus on question generation. The authors note that it is important to remove easy questions and refined question strategies to avoid producing silly or obvious questions.

Shijie Zhang et al.[7] describes a system for generating questions, in the context of learning, which also comprises the NLP components of lexical processing, syntactic processing, logical form, and generation. This system uses summarization as a pre-processing for identifying information about asking a question. The authors selecting questions created by Content QA Generator is difficult. Guokun Lai,[8] describes a system to generate factoid questions automatically from large text. User questions matched against these pre-processed factoid questions in order to identify relevant answer in a Question -Answering system.

Xinya Du et al, the task of question generation is defined as the automatic generation of questions from various input sources. Sources can be raw text, a database or some form of semantic representation. They further argue that the "goodness" of a question can only be determined by looking at the context the question was posed in [9].

Unnat Jain et al. focus on the problem of removing words from a sentence to create fill-in-the-blanks quizzes for language learning. For the removed words they create distractors and evaluate them in terms of reliability and validity.

Manuscript received on February 10, 2020.

Revised Manuscript received on February 20, 2020.

Manuscript published on March 30, 2020.

* Correspondence Author

Animika, Scholar, Department of computer Engineering, Arya College of Engineering, Kukus, Jaipur, Rajasthan, India.

Dr. Vibhakar Pathak, Professor, Department of computer Engineering, Arya College of Engineering, Kukus, Jaipur, Rajasthan, India.

Dr. Vishal Shrivastava, Professor, Department of computer Engineering, Arya College of Engineering, Kukus, Jaipur, Rajasthan, India.

Dr. Akil, Professor, Department of computer Engineering, Arya College of Engineering, Kukus, Jaipur, Rajasthan, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Questions Generation for Reading Comprehension using Coherence Relations

A distractor has to be reliable; meaning that it cannot be replaced with the answer, thereby avoiding multiple correct answers to a question [10].

Manish Agarwal et al. deals with the technique for generating questions focused on Named Entities (NE), temporal or position information, and semiconductor-based functions correlated with terms in input sentences.

The question recognizer module checks whether a specific question type can be generated from a clause in each input sentence, and also identifies the possible cue phrase in the question type clause. The generator module replaces the cue expression with the query term and reorder the clause pieces to ensure grammatical correctness[15].

III. PROPOSED QUESTION GENERATION METHOD

The main goal of this thesis is to grip the structured information from the knowledge base to design meaningful questions and answers to improve reading comprehension and learning capability. Therefore, the task comprises the selection of the question's content, meaning which clues are contained in the question, and the question's answer. We decided to choose a structural query representation as to the preliminary formulation of the question. Using this representation of the query enables us to develop a method to express it into natural language, which is required for users to interpret the system's output.

Steps 1 Data set Preparation

Data set : we use reading comprehension data set that related to history, science, news, article domain. For training and testing reading comprehension. after retrieving data set we need a different paragraph in order to generate a question and comparison between and our system.

Steps 2 Paragraph Selection:

We download articles from the corpus, to classify the article's state, the amount of sentences in an article is more than 5, we pick papers with more than 7. When an article is very small, the likelihood of detecting coreference is poor. Generally, when those two are far from each other, it is difficult to find individual coreference between two organizations.

Steps 3 Apply Rhetorical Structure Theory for Coreference Detection:

RST are characterized by three parameters: the nucleus, the satellite and the interaction between the nucleus and the satellite. The nucleus is an action and satellite either describes this action. Here the discourse graph associated with the document is input to the system, which in turn extracts all relevant nucleus-satellite pairs.

Table- I: Relation set

Relation set	Offered From
Performance, (N,S)	Goal (n) Action(s)
Patternsation, (N,S)	Pattern (n), concept or algo(s)
Execution (N,S)	Plan/agent (N), accomplishment /goal(s)
Enablement, (N,S)	Devices (N), Action (S)

We discard a sentence when it contains the most representative entity because a question generated from that type of sentence does not require multiple sentences to answer.

Each pair is represented As the tuple: Relation (Nucleus,Satellite). Prior to applying any syntactic transformations on the text spans, we remove all leading and/or trailing conjunctions, adverbs and infinitive phrases from the text span. Further,if the span begins or ends with transition words or phrases like 'As a result' or 'In addition to', we remove them as well.

Steps 4 Text-span Identification

We associate each text span with a Type depending on its syntactic composition. The assignment of Types to the text spans is independent of the coherence relations that hold between them.

Table- II: Span Types with relevant examples

Span Type	Characteristic of Span
Type 0	A group of many sentences
Type 1	One sentence, or a phrase or clause not beginning with a verb, but containing one
Type 2	Phrase or clause beginning with a verb
Type 3	Phrase or clause that does not contain a verb

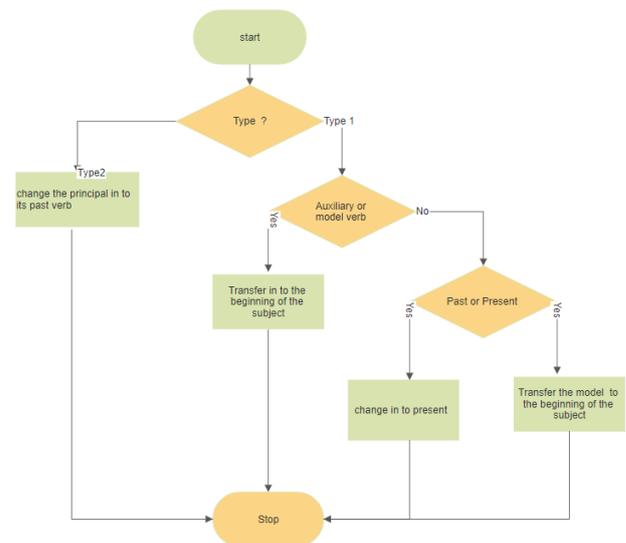


Figure 1. Text Spans Syntax Identification and transformations

Steps 5 Text Spans Syntax transformations

If the text span is of Type 1 or Type 2, we analyze its parse tree and perform a set of simple surface syntax transformations to convert it into a form suitable for QG. We first use a dependency parser to find the principal verb associated with the span, its part-of-speech tag and the noun or noun phrase it is modifying. Then, according to the obtained information, we apply a set of syntactic transformations to alter the text. No syntactic transformations are applied on text spans of Type 0 or Type 3.



We directly craft questions from text spans that belong to these Types.

Table III: Relation Template.

Relation	Template type0	Template type1	Template type2	Template type3
Performance, (N,S)	[Nucleus] What led to the start of action ?	Why [Nucleus] ?	What [Nucleus] ?	What caused [Nucleus]?
Patternsation, (N,S)	[Satellite]. What leads to formation of concept ?	Why [Satellite] ?	What [Nucleus] ?	What caused[Nucleus]?
Execution (N,S)	[Nucleus]. What leads to the accomplishment ?	Why [Satellite] ?	What [Nucleus] ?	What caused [Satellite]?
Enablement, (N,S)	[Nucleus]. What led to action ?	Why [Nucleus] ?	What [Nucleus] ?	What caused [Satellite]?

Steps 6 QuestionGeneration

we obtain a text form suitable for QG. A template is applied to this text to formulate the final question. Table defines these templates. The design of the chosen templates depends on the relation holding between the spans, without considering the semantics or the meaning of the spans. This makes our system generic and thereby scalable to any domain.

IV. EXPERIMENTAL RESULTS QUESTION EVALUATION

The first thing we noticed was the high percentage of grammatical and semantically relevant queries. Intuitively, it certainly sounds very dangerous to remove a priori uncertain parts of a sentence and inject them into predefined models whose grammar and semantics may or may not be compatible, but as we have seen, we can produce several grammatical and semantically correct questions with some very basic filters and modifiers.[16]

To evaluate the quality of generated questions,we used a set of criteria that are defined below.We considered and designed metrics that measure both the correctness and difficulty of the question.All the metrics use a two-point scale: a score of 1 indicates the question successfully passed the met-ric, a score of 0 indicates otherwise.

Table IV: Average score for the Question Appropriateness: evaluation.

Evaluation System	System	Performance	Patternsation	Execution	Enablement	Average
Grammatical	mh	0.95	0.94	0.91	0.87	0.915
	qg	0.92	0.94	0.91	0.90	0.925

Semantic	mh	0.95	0.91	0.97	0.88	0.8923
	qg	0.95	0.92	0.97	0.89	0.9325
Superfluous	mh	0.84	0.81	0.77	0.82	0.81
	qg	0.90	0.82	0.82	0.83	0.8425
Question appropriateness	mh	0.93	0.83	0.95	0.75	.865
	qg	0.92	0.91	0.94	0.83	0.9

Table 5.4: Average score for the Question Appropriateness: evaluation.

• Grammatical correctness of questions:

This metric checks whether the question generated is only syntactically correct. We do not take into account the semantics of the question.[17]

• Semantic correctness of questions:

We account for the meaning of the generated question and whether it makes sense to the reader.It is assumed if a question is grammatically incorrect, it is also semantically incorrect.

• Superfluous use of language: generated questions may contain information not required by the student to arrive at the answer.

• Question appropriateness: This metric judges whether the question is posed correctly or not.

V. RESULTS DISCUSSION

We decided to test our proposed QG strategy in a manner that was conscious of intent to produce the queries. Evaluation systems of prior QG projects centered on acceptability. We have seen that both domain-specific and general-purpose models may have learning-value problems, so combining the two is essential. We have compared our proposed QG strategy with Heilman and Smith MH.

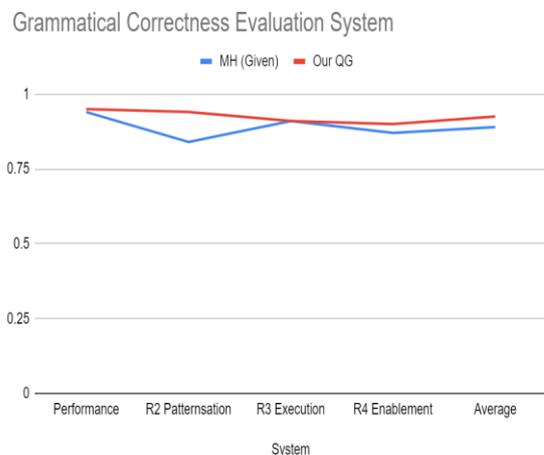


Figure 2. Grammatical Correctness evaluation.

Questions Generation for Reading Comprehension using Coherence Relations

In Figure 2. we checked the Grammatical Correctness between our proposed Qg method given mh method. To specify our results we use type of relation (Performances, Patttensation Execution,Enablement) as x axis and y axis represented the range of efficiency between 0 to 1. When we show the graph value of our proposed method is high compare than to The mh method in Grammatical Correctness. When we compare the average of both methods. Our QG method (0.89)also be higher than MH method (0.925).

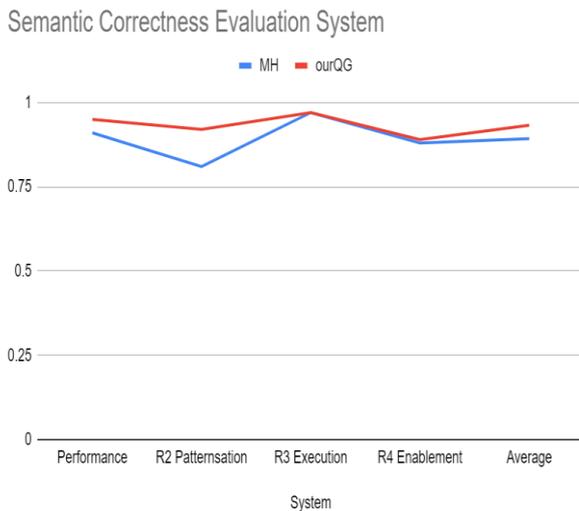


Figure 3. Semantic Correctness evaluation .

In Figure 3 . we checked the Semantic Correctness between our proposed Qg method given mh method. To specify our results we use type of relation (Performances, Patttensation Execution,Enablement) as x axis and y axis represented the range of efficiency between 0 to 1. When we show the graph value of our proposed method is high compare than to The mh method in Semantic Correctness. When we compare the average of both methods. Our QG method (0.9325)also be higher than MH method (0.8935).

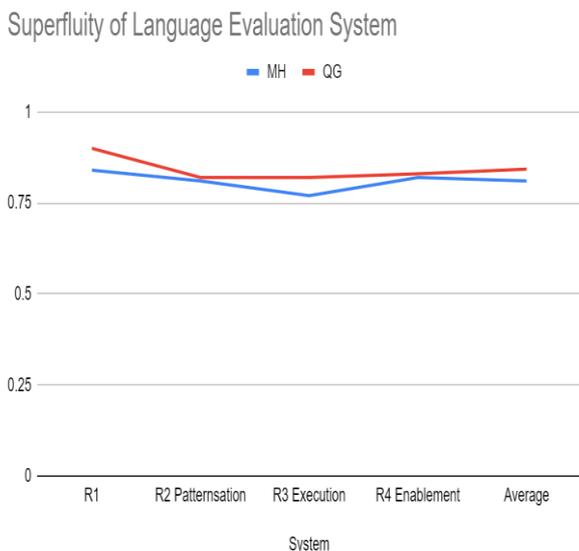


Figure 4. Superfluity of Language evaluation .

In Figure 4 . we checked the Superfluity of Language evaluation between our proposed Qg method given mh method. To specify our results we use type of relation (Performances, Patttensation Execution,Enablement) as x axis and y axis represented the range of efficiency between 0 to 1. When we show the graph value of our proposed method is high compare than to The mh method in Superfluity of Language evaluation. When we compare the average of both methods. Our QG method (0.8425)also be higher than MH method (0.8132).

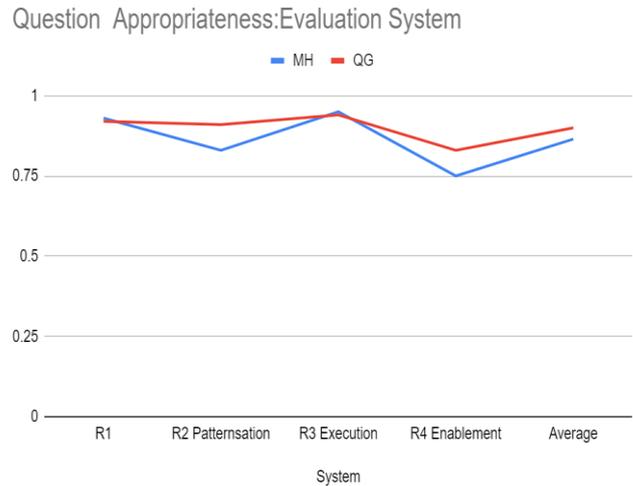


Figure 5. Question Appropriateness evaluation.

In Figure 5 . we checked the Question Appropriateness evaluation between our proposed Qg method given mh method. To specify our results we use type of relation (Performances, Patttensation Execution,Enablement) as x axis and y axis represented the range of efficiency between 0 to 1. When we show the graph value of our proposed method is high compare than to The mh method in Question Appropriateness evaluation. When we compare the average of both methods. Our QG method (0.90)also be higher than MH method (0.865).

VI. CONCLUSION AND FUTURE SCOPE

In This paper I have presented a novel approach to generating questions from text that combines the relation flexibility template based on different GQspecific with the of question.categorizing current GQ models summarizing three emerging trends: multi-task learning, wider input modalities, and deep question generation. templates have each been used by other approaches to QG.[18]

In our Approach we have begun experimenting with a multi-sentence QG method that design templates, a glossary, and discourse connectors. Inter-sentential discourse connectives,such as for example, therefore, however, and furthermore, provide an inexpensive and reasonably robust way to identify groups of sentences that we can and should use to generate questions .

We demonstrate a system that uses discourse connectors for multi-sentence QG, but their approach does not truly integrate multi-sentence content into questions.

Once they identify the connective arguments, they use syntactic transformations to produce questions from one and only one of those arguments. Generated questions in particular domain are at the level of understanding. There is much more to explore in terms of algorithms and evaluation. Our approach generated a set of templates and questions attempt. Although more work can be done to improve the quality of generated questions like general dialog management, no research has explored when machines should ask engaging dialog questions. Modeling asking questions as an interactive and dynamic process can become an interesting topic ahead. QG with interface simulation in dialog or suggestion framework has not yet been investigated. Modeling user condition and knowledge specifically takes us to customized QG, which dovetails extreme, end-to-end QG with deep user modeling and combines the dual level–understanding much in the same way as in vision–image generation[19].

REFERENCES

1. Takshak Desai, Parag Dakle, and Dan Moldovan. 2018. Generating questions for reading comprehension using coherence relations. In The 5th Workshop on Natural Language Processing Techniques for Educational Applications (NLP-TEA@ACL), pages 1–10.
2. Xinya Du and Claire Cardie. 2018. Harvesting paragraph-level question-answer pairs from wikipedia. In Annual Meeting of the Association for Computational Linguistics (ACL), pages 1907–1917.
3. Han Guo, Ramakanth Pasunuru, and Mohit Bansal. 2018b. Soft layer-specific multi-task summarization with entailment and question generation. In Annual Meeting of the Association for Computational Linguistics (ACL), pages 687–697.
4. Zhihao Fan, Zhongyu Wei, Siyuan Wang, Yang Liu, and Xuanjing Huang. 2018. A reinforcement learning framework for natural question generation using bi-discriminators. In International Conference on Computational Linguistics (COLING), pages 1763–1774.
5. Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. In Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 3901–3910.
6. Vishwajeet Kumar, Kireeti Boorla, Yogesh Meena, Ganesh Ramakrishnan, and Yuan-Fang Li. 2018a. Automating reading comprehension by generating question and answer pairs. In The Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), pages 335–348.
7. Shijie Zhang, Lizhen Qu, Shaodi You, Zhenglu Yang, and Jiawan Zhang. 2017. Automatic generation of grounded visual questions. In International Joint Conference on Artificial Intelligence (IJCAI), pages 4235–4243.
8. Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard H. Hovy. 2017. RACE: large-scale reading comprehension dataset from examinations. In Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 785–794.
9. Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In Annual Meeting of the Association for Computational Linguistics (ACL), pages 1342–1352.
10. Unnat Jain, Ziyu Zhang, and Alexander G. Schwing. 2017. Creativity: Generating diverse questions using variational autoencoders. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 5415–5424.
11. Mitesh M. Khapra, Dinesh Raghu, Sachindra Joshi, and Sathish Reddy. 2017. Generating natural language question-answer pairs from a knowledge graph using a RNN based question generation model. In Conference of the European Chapter of the Association for Computational Linguistics (EACL), pages 376–385.
12. Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2017. Neural question generation from text: A preliminary study. In CCF International Conference of Natural Language Processing and Chinese Computing (NLPCC), pages 662–671.
13. Xinya Du and Claire Cardie. 2017. Identifying where to focus in reading comprehension for neural question generation. In Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 2067–2073.

14. Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: visual question answering. In IEEE International Conference on Computer Vision (ICCV), pages 2425–2433.
15. Manish Agarwal, Rakshit Shah, and Prashanth Mannem. Automatic question generation using discourse cues. In Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications, pages 1–9. Association for Computational Linguistics, 2011.
16. Michael Heilman. 2011. Automatic factual question generation from text. Language Technologies Institute School of Computer Science Carnegie Mellon University, 195.
17. Husam Ali, Yllias Chali, and Sadid A Hasan. 2010. Automation of question generation from sentences. In Proceedings of QG2010: The Third Workshop on Question Generation, pages 58–67.
18. Michael Heilman and Noah A Smith. Good question! statistical ranking for question generation. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 609–617. Association for Computational Linguistics, 2010.
19. Michael Heilman and Noah A Smith. Extracting simplified statements for factual question generation. In Proceedings of QG2010: The Third Workshop on Question Generation, pages 11–20, 2010.

AUTHORS PROFILE



Animika, M.Tech Scholar :Department of computer Engineering Arya College of Engineering ,kukus,jaipur,Rajasthan.



Dr. Vibhakar Pathak, Professor, has 22 years of research Experiences. He head published many research papers His area of research include cloud computing image processing, and robotics.



Dr. Vishal Shrivastava Professor, has 16 years of research Experiences. He head published more than 200 research papers in renowned conferences and journals His research interest include data mining, Ai and Network security.



Dr. Akil Professor, has 22 years of industry and research Experiences. He head published many research papers in renowned conferences and journals His research interest include computer network ,algorithms and real time system.