International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-5, March 2020

Automatic Question Generation using Sequence to Sequence RNN Model

Alina K. Jayarajan, Ajumol P. A., Ani Sunny

Abstract*: Automatic Question Generation (AQG) has recently received growing focus in the processing of natural language (NLP). This attempts to create questions from a text paragraph, where certain sub-spans of the passage in question will answer the questions produced.. Traditional methods predominantly use rigid heuristic rules to turn a sentence into related questions. In this research, we suggest using the neural encoder-decoder model to produce substantive and complex questions from the sentences of natural language. We apply a attention-based sequence to sequence learning paradigm for the task and analyze the impact of encoding sentence vs. knowledge at paragraph level. Information retrieval and NLP are the core components of AQG. It incorporates the application of production rules; recurrent neural network (RNN) based encoder-decoder sequence to sequence (seq2seq) models, and other intelligent techniques. RNN is used because of its long short term memory power (LSTM).The proposed system focus on generating factual WH type questions.

Keywords: Long-Short Term Memory, Natural Language Processing, Recurrent Neural Networks,

I. INTRODUCTION

Natural Language Processing (NLP) handles the communication between machines and humans [1]. A decisive aim of NLP is to peruse, view, comprehend, and understand human languages in a meaningful manner. Many NLP procedures rely on machine learning to decide the significance of human dialects. Improving NLP implementations is difficult because machines usually require humans to “sing” to them in a precise, unambiguous and excessively ordered programming dialect, or through a limited number of clear voice commands. Current approaches to NLP are based on deep learning [2], a sort of AI that looks at and employments designs in information to move forward a program understands. Deep learning models require massive amounts of labeled data to train and define important associations, and the integration of this sort of big data collection is currently one of the major hurdles for NLP.

Earlier solutions to NLP included a more rule-based approach; in which simpler machine learning algorithms were instructed the words and phrases were offered when those phrases tended to search for in text and comprehensive responses. But deep learning is a more powerful, intuitive method in which algorithms learn to understand the context of speakers from many instances, almost like how a child might know the human language. The benefit of natural language processing can be seen by speaking about the two following articulations:”The security of cloud computing should be a feature of any service level agreement” and” A perfect SLA offers a smoother night’s rest-even in the internet.” If you are using national language search tools, the system will define cloud computing as an entity, cloud as an abbreviated form of cloud computing, and SLA as a service level agreement industry acronym. These are the kinds of vague elements sometimes found in the human language and that machine learning algorithms have been historically poor in interpretation. Today algorithms can accurately perceive them, with developments in deep learning and artificial intelligence. This has consequences for the types of data that can be analyzed. Online awareness is being produced more and more every day, and much of that is a natural human language. Until recently businesses were unable to analyze the results. But advances in the NLP allow research and learning from a wider array of sources of data.

Deep, neural-based methods to learning showed significant performance gains in many artificial intelligence tasks. Nevertheless, the complex structure of those networks also renders understanding their forecasts complicated. This observation is expressed by attention based sequence-to-sequence models (seq2seq) also referred to as encoder models [3]. Seq2seq models demonstrated state-of -the-art performance across a wide range of applications, including machine translation, natural language creation, image captioning, and definition.. Recent results show that these models demonstrate human-level performance in machine translation for certain important domains.

Seq2seq models are useful because they provide an effective guided approach to processing and predicting sequences without requiring the source-to-target relationships to be manually precise. These systems learn how to reorder, transform, compress, or expand a source sequence to an output goal sequence, first using a large internal state representation, and then decode the source sequence using a single model model. These models provide a general purpose framework, with sufficient data, for learning how to predict sequences.

Revised Manuscript Received on March 30, 2020.
* Correspondence Author
Alina K. Jayarajan*, Department of Computer Science and Engineering, Mar Athanasius College of Engineering, Kothamangalam, Kerala, India. E-mail: alinajayarajank@gmail.com
Ajumol P. A., Department of Computer Science and Engineering, Mar Athanasius College of Engineering, Kothamangalam, Kerala, India. E-mail: ajumolantony0203@gmail.com
Ani Sunny, Department of Computer Science and Engineering, Mar Athanasius College of Engineering, Kothamangalam, Kerala, India. E-mail:anisunny88@gmail.com

While the effect of the seq2seq models was obvious, questions arise about the added complexity and instability of models based on deep learning.

These models act as black-boxes during estimation, making it hard to track the source of the errors. The high-dimensional internal representations complicate model analysis as it changes outcomes. While deep learning shares this property, language-involving errors are often very clear to human readers[4]. For example, the incorrect implementation of a seq2seq translation program converting ”good morning” into ”strike them” culminated in a widely publicized event leading to wrongful arrest. Popular but troubling failures in seq2seq models involve machine translation systems that effectively mistranslate a sentence, image captioning systems that create an incorrect caption, or speech recognition systems that generate an incorrect transcript. Automatic questions production is part of Natural Language Processing (NLP) [1]. It is a research area where many scholars introduced their findings and is still an area under study in order to achieve greater accuracy. Several researchers worked with NLP in the area of automated query generation, and several strategies and models were established to produce the various types of questions.

Teachers / professors / tutors (academics) nowadays spend a great deal of time manually producing test papers and quizzes. Similarly, students spend a lot of time analyzing themselves (self-calibration). In addition, for self-analysis students are reliant on their mentors. Therefore, we are focusing on this field of the NLP, which currently has a huge scope for growth. We want to create a code application system which can help you calibrate yourself and delete any tutor dependencies. Here, students can give the input text of whatever material they referred to, and on that basis they get a set of questions with answers from which they can do an analysis of themselves (self-calibration). Mentors employ a similar approach to creating test papers and quizzes.

In addition, online tests have become very common, including many big reviews such as GATE, CAT, and NET. Multiple Choice Questions (MCQ) is very simple to assess and its assessment is carried out via computerized applications so that outcomes can be announced within a few hours, and the evaluation process is 100.By making this computerized application, we can lower an educator’s task. A great deal of time can be saved if we know what appropriate questions can be asked for the given text input. So we want to develop a system that can produce specific logical questions from the input of the given document. Only humans are capable of achieving this right now. This paper proposes a program which can produce various logical questions from the feedback of the provided document. Question creation process is based on artificial intelligence technologies and natural language processing skills. The potential benefits of using automated systems to generate questions help reduce people’s dependence on questions and other needs related to interactive systems.

A. Challenges in Question Generation

We categorize QG challenges broadly into four categories: lexical challenges, syntactic challenges, discourse-related challenges and other challenges associated with using QG tools

- **Lexical Challenges**: The solution to a question’s semantics affects the form of the question. The answer decides in turn whether the question is going to be a who question, a where question, etc. In QG, the difficulty is that queries preferably show some variance from the content of the text. Often, one might want to produce a number of wordings for a particular question.
- **Syntactic Challenges**: QG systems need to mold some of the complexities of natural language syntax to generate questions: For example, they need to do things such as identify phrasal boundaries (e.g., which words are part of a response), identify predicates and their arguments, and extract information from nested structures such as relative clauses.
- **Discourse Challenges**: Texts are usually written as complete discourses that stand by themselves. Where new entities or concepts are introduced, Usually writers use very specific descriptions, such as Abraham Lincoln- at least in the kinds of information texts that we focus on. Nevertheless, individual sentences build on earlier sentences and authors use language tools such as anaphora to avoid repeating the text. Consequently, as a single sentence or clause can rely on its sense of debate, it can contribute to ambiguous and awkward questions by separating facts from their meanings and raising questions on them.
- **Challenges Related to Use of Question Generation Tools**: Not all questions are value-equivalent. The value of a question also depends on the particular context in which it is to be used. For example, a teacher whose students read a text about Dublin may choose either to focus on basic factual details about the city (e.g., What is ublin’s capital?), quite specific data (e.g., In what year did Vikings create Dublin’s city?), or various other types of knowledge. In addition, an instructor may want to focus on specific elements of a text as they relate to other subjects in his or her curriculum. A possible approach to assessing information’s significance and value could be to apply NLP methods to sum up extractive data.

II. RELATED WORKS

Several studies have shown that most students find it difficult to recognize their own gaps in knowledge and pose very few questions [4]. Questions are useful in understanding the learners ’ deficits in comprehension and in enhancing their learning. When you want students to read a literature review or write an essay, this often means developing not only critical communication skills, but also the ability to obtain them(i.e. citing sources as evidence to support their arguments) and Combining information to comprehend and analyze multiple documents (i.e. presenting evidence in a clear and convincing way).

Reynolds and Bonk[6] find that a group of students with uniform trigger questions perform better than those students who get no triggering written questions.However, these problems are too common, and it is doubtful that they will give strong help when writing on a particular topic. Many content-related questions need to be asked and in the process of providing feedback to the students, most educators would ask those questions.

Many AQG systems[3][6] depend on the text-to-question function in the Automatic Question Generation (AQG) field where a series of content-related questions are created on the basis of a specific document.

The answers to questions you have generated are usually found in the text.Initially, Heilman and Smith [4] developed an AQG mechanism for generating factual issues with an overgenerating and ranking methodology based on natural language processing methods such as the Name Object Recognizer and Wh-movement Rules, as well as a comparative rating system for feature-based problems. Such approaches ' main uses include literacy understanding and language testing which may not suit academic learning.

A variety of relationships can be drawn between query generation (QG) and other NLP fields, and computer linguistics[5]. Our example of QG can be interpreted as an instance of text-to-text monolingual creation, since both input and output are natural language text. The processing of text to text has been of great interest to NLP researchers throughout the last few years, with substantial work on issues such as word compression, paraphrase production and text simplification[7].

QG faced with a number of fascinating new problems in this field[6]. Connections may also be made for answering questions, producing natural language and even translating computers. We speculate that approaches to generate-and-rank[9]such as ours might be effective for other problem-solving text. In the over generate portion of a program one can encode complicated linguistic information in the form of rules. While this capacity to encrypt knowledge may not be as important for issues such as machine translation, where large usable input and output sentence repositories are available[10], it may be very useful for issues such as the development of paraphrases where relevant data is scarce. Overgenerate-and-rank approaches[7] often allow one to use machine learning to optimize outputs, and our approach to gathering a customized dataset of human-rated outputs provides a relatively easy way to develop a statistical ranker[11]. QG also provides inspiration for the work on key NLP methods (e.g., syntactic parsers, designated individual recognizers, and co-reference resolvers). These methods offer valuable theoretical analyzes for input sentence translations into queries. These tools make mistakes, so if those methods changed, QG programs would also change.

RNNs [8][12] may be used as a language model to predict future elements of prior sequence elements given in a series. Nonetheless, there is still a shortage of components needed to build translation models, as we can only act on a single sequence, while translation operates on two sequences–the input sequence and the converted sequence.

Build sequence to sequence models [3] by adding an encoder stage and a decoder step on top of the language models. In the encoder stage, a model transforms a sequence of inputs into a defined representation (like a English sentence). In the decoder stage the output sequence (such as the translated sentence) as well as the encoder's fixed representation will be trained on a language model [13]. Given that the decoder model uses both a binary representation of the input sequence and the expression sequence, it can make smarter decisions regarding possible terms based on the current term. For example, in a standard language model, we might see the word "crane," and not be sure if the next word should be about the bird or heavy machinery [14]. However, the decoder may remember that the sequence of inputs was about architecture, not flying animals, unless we transfer an encoder background also. The decoder may select the appropriate next word and, given the context, provide more precise translations.

III. SYSTEM ARCHITECTURE

Most AQG applications can be broadly classified into two main groups based on their objectives: 1) Fostering development of dialog and interactive question and answer systems, and 2) Automating education evaluation. Questions are either generated directly from the exhibitory texts or after the domain topic has been identified. Now that we've introduced the issue of generating factual questions and described the challenges it poses, We present our solutions (fig.1) in stage 1 of our system, transforming a sentence or a set of phrases from the text into a simple declarative statement by (optionally) altering or transforming lexical objects, syntactic structure and semantics..Our architecture includes extraction operations and dynamic simplification of sentences and pronoun resolution. Future work may adapt other NLP transformations, such as sentence compression, sentence fusion, or paraphrase creation, for use in this level.

In stage 2, by introducing a sequence of well-defined syntactic transformations (WH-movement, subject-auxiliary inversion, etc.), the declarative sentence becomes a set of questions. We start by dividing the selected sentences into simple and complex phrases, each of which is processed separately. The sentences which include discourse connections are classified as dynamic phrases. In this step, the basic sentences are split into English sentence subsections i.e. Element, object, verb. Named Entity Recognizer(NER) is then analyzed to classify its coarse representation over the sentence subject and object. The NER then specifies the tagged type of word as Person / Human, Location, and Organization. The coarse ones are classified as follows:

- Human: It involves the a person's name. Entity: It covers creatures, trees, lakes, etc.
- Time: This is going to be any time, date or period like year, Monday, 9 a.m., last week etc.
- Location: These will be the words representing locations like country, town, school, etc.
- Count: This class will hold all of the counted elements, such as 9 men, 7 workers, weight and size measurements, etc.
- Organization: organizations that include businesses, institutes, government, market, etc.

Once the words are assigned to coarse groups in the sentence, we call the relation between the words in the sentence. For example, if the statement has the "Human Verb Human" form, it will be classified as the "who and how" forms of queries. If a preposition representing the date occurs, then we add the question term "Why" to its definition. Then they are categorized according to the different rules, depending on the form of the sentences and the sentences. Sentences which include connective language, i.e. conjunctions such as when, for example, complex sentences are known then etc. Connectives such as and/or demonstrating conjunction interactions were not find to be good candidates for producing wh-type questions.

Our system also pays attention to improving the quality of the sequence-to-sequence RNN questions produced. We want to implement it using encoder-decoder networks (fig.2), which will improve the quality drastically[9].For its neural machine translation (language translation) and recurrent neural networks, Google used encoder-decoder networks. We also use Stanford Parser and NLTK to help with grammar analysis and more basic natural language processing by keeping the core encoder-decoder [10]. The input is given as text to Seq2Seq RNN Model and output is also text. It is used in machine interaction and machine translation. THere are four parts, Input Layer, Embedding Layer, LSTM Layer and Production Decoder Layer. The input sequences are transformed into two vectors of the body. Train LSTM to get the destination production. Recalling the previous words and creating the question has more strength.

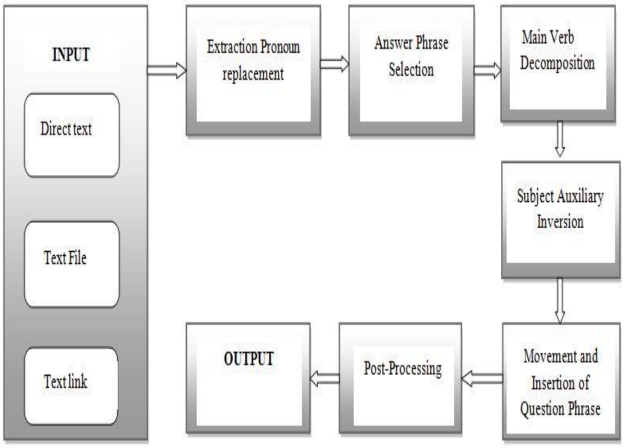

Fig.1.Proposed System Architecture

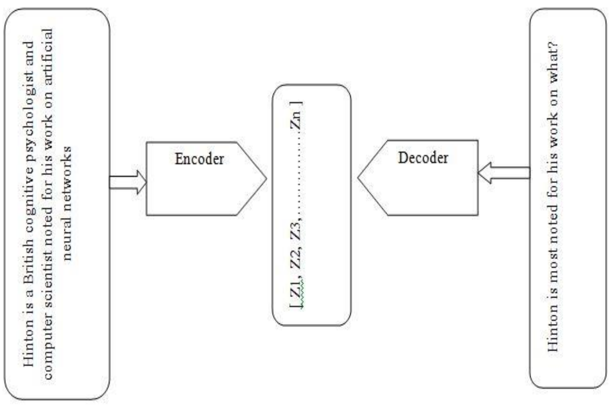

Fig.2. Encoder-Decoder Internal View

A. Ranking Question Importance

A question generation system can optimize its effectiveness by ranking performance question to determine can issues are more likely to be relevant. Heilman and Smith used a database ranker for logistic regression, which concentrated on linguistic accuracy. In the top 20 percent of questions produced, the ranker more than doubled the percentage of appropriate questions, from 23 percent to 49 percent (2011).Others sought the logistic regression strategy too, but with less performance. One program was able to identify that with 86 percent accuracy a problem was not acceptable; However, the annotator found 83 percent of the questions inappropriate, so the utility of the classifier is not apparent.

Since our proposed system produces usually correct grammatical questions, we agreed to test the importance of the questions, a criteria that is often ignored. To this end we used the algorithm TextRank for keyword extraction. For a given input passage TextRank listed the top 25 nouns. Then each question generated received a score based on the percentage of the top TextRank words it included, with a penalty for very short questions like What is the keyword? Our evaluation found that the performance of important questions often raises their scores for acceptability.

IV. RESULT AND ANALYSIS

Software configuration is the application framework that influenced the project. It is important to choose the language which is suitable for the project. Also the operating system should be chosen which will be appropriate for the project. These selections play a significant role in the performance of the developed tool. First we need to set up the Docker container for implementing the proposed method. These selections play a significant role in the performance of the developed tool. First we need to set up the Docker container for implementing the proposed method. The performance of the developed system is compared to the existing system. Scoring mechanism is used to evaluate system accuracy. Scoring is also called prediction, and is the process of generating values based on a trained model of machine learning, given some new input data. The values or scores that are created can represent predictions of future values, but they might also represent a likely category or outcome. The meaning of the score depends on the type of data you provide, and the type of model that you created. In machine learning scoring is widely used to mean the process of generating new values, given a model and some new input. Analysis of inacceptable issues indicated sources of error as well as places for future work. Language idiomatic was liable for a few mistakes. For eg, the phrase: Few members spend time in the chamber other than when they talk or vote prompted the problem to arise: What are few members spending? In this scenario time is the direct object grammatically, which is why this question was created because spending time is an idiom. Another way to avoid increasing this question would be to look out related idiomatic phrases and rephrase them, rendering the idiomatic vocabulary more clear. A further difficulty is that some models work on different topics differently than others. For examples, a prototype that fits the S-Vattr pattern is How would you characterize the subject matter? That raised the question: How can a gland be characterized? Through reply: a structure made up of one or more cells modified to synthesize and secrete chemical substances: How do you characterize the ocean? From the phrase: Once the sea had become the world's fourth largest water source. Techniques must be used to classify noun phrases appropriate for interpretation problems, a challenge to be discussed in future work Another issue is that pre-processing is inadequate to remove sentences such as: The question-creating figure shows related monetary policy episodes: Where are the specific monetary policy episodes listed? Our machine pre-processing unit eliminates most of the references to statistics and tables but not all.

Another concern is that it describes a sequence of events in words, in which case in turn, a given sentence may be unclear. Political authority with the government seemed to function as never before, raising the question: What did the political authority seem to be doing? The question remains ambiguous and unachievable.. This problem indicates that for certain topics that are not accessible in general-purpose question generators, versatility is necessary. Nevertheless, the essence of a system for producing a general-purpose application, as opposed to a process for specific topics or sources, is fundamentally contradictory.

Input:

```
./get_qnas "Waiting had its world premiere at the \
  Dubai International Film Festival on 11 December 2015 to positive reviews \
  from critics. It was also screened at the closing gala of the London Asian \
  Film Festival, where Menon won the Best Director Award."
```

Output:

```
who won the best director award ? menon -2.38472032547
when was the location premiere ?  11 december 2015  -6.1178450584412
```

Fig. 3. Sample Output

V. CONCLUSION

The generation of automatic questions is a part of Natural Language Processing (NLP). It is a research area in which many researchers have presented their work and is still an area under investigation to achieve greater accuracy. The proposed method will help a person produce query automatically from the provided document. It is a method in which it generates reasonable questions from the data as information, providing an input text to the device. The potential benefits of using automated systems to generate questions help reduce dependence on humans to generate questions and other needs related to systems that interact with natural languages. Question generation process is based on the capabilities of artificial intelligence and Natural language processing. Each AQG framework has different criteria for evaluating the consistency of queries. Here we ensure the quality of the questions by using sequence to sequence recurrent neural network. Because of its Long Short Term Memory power.

REFERENCES

1. Lotfi A. Zadeh' The University of California, USA Precisiated Natural Language (PNL)-toward an Enlargement of the role of natural languages in Computation, deduction, definition and decision
2. Jinmiao Chen ; N.S. ChaudhariImprovement of bidirectional recurrent neural network for learning long-term dependencies Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.
3. Masato Mimura, Sei Ueno, Hirofumi Inaguma ; Shinsuke Sakai ;
4. Tatsuya KawaharaLeveraging Sequence-to-Sequence Speech Synthesis for Enhancing Acoustic-to-Word Speech Recognition2018 IEEE Spoken Language Technology Workshop (SLT)
5. Heilman and Smith "Automatic question generation system". Proceedings of the Institution of Civil Engineers: Transport, 158, (3), 149-155, 2005.
6. Graesser, A.C., Person, N.K.: Question asking during tutoring. American Educa- tional Research Journal 31 (1994) 104137
7. Reynolds, T., Bonk, C.: Computerized prompting partners and keystroke recording devices: Two macro driven writing tools. Educational Technology Research and Development 44(3) (1996) 83
8. Palubinskas, G., Kurz, F., and Reinartz, P., 2009. "Domain specific automatic Chinese multiple-type question generation
9. Lucy Vanderwende. 2008. The importance of being important: Question generation. In Proceedings of the 1st Workshop on the Question Generation Shared Task Evaluation Challenge, Arlington, VA.
10. Michael Heilman. 2011. Automatic factual question generation from text. Ph.D. thesis, Carnegie Mellon University
11. Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into texts. Association for Computational Linguistics.
12. CHEN Wenjie, CHEN Lifeng, CHEN Zhanglong, TU Shiliang, "Automatic formation of questions and answers on the basis of the knowledge base," in Proc. IEEE ICPPW '05, Oslo, Norway, pp. 258 – 264, June 2005
13. Ming Liu, Rafael A Calvo, and Vasile Rus. 2010. Automatic question generation for literature review writing support. pages 45–54..
14. Karen Mazidi and Rodney D Nielsen. 2015. Leveraging multiple views of text for automatic question generation. In Artificial Intelligence in Education, Springer LNCS.
15. Siuli Roy, Somprakash Bandyopadhyay, Munmun Das, Suvadip
16. Batabyal,, Sankhadeep Pal,"Computational Intelligence Framework for Automatic Quiz Question Generation", LAP Lambert Academic Publishing

AUTHORS PROFILE

Alina K. Jayarajan, received Bachelor of Technology in Computer Science and Engineering from Govt. Engineering College , Idukki in 2017 and currently pursuing Master of Technology in Computer Science and Engineering from Mar Athanasius College of Engineering, Kothamangalam affiliated to APJ Abdul Kalam Technological University. Her research interest is in Artificial Intelligence and Natural Language Processing.

Ajumol P. A., received Bachelor of Technology in Computer Science and Engineering from KMEA Engineering College, Edathala, Aluva in 2018 and currently pursuing Master of Technology in Computer Science and Engineering from Mar Athanasius College of Engineering, Kothamangalam affiliated to APJ Abdul Kalam Technological University. Her research interest is in Computational Intelligence and Natural Language Processing.

Ani Sunny, is currently working as assistant professor in the Department of Computer Science and Engineering, Mar Athanasius College of Engineering,Kothamangalam, Kerala, India. She received her B.Tech degree in 2011 in Computer Science and engineering from Mahatma Gandhi University, Kottayam and M.Tech in 2014 in Computer Science and Engineering. She is interested in the area of Hardware and Microprocessors, Networks and Image Processing