EasyChair Preprint

№ 10299

Multiple Choice Question Generation Using BERT XL NET

S.Adi Lakshmi, Rajesh Saturi, Anupriya Bharti, Meghana Avvari and Battu Bhavana

May 30, 2023

Multiple choice Question Generation Using BERT XL NET

S.Adi Lakshmi[1], Assistant Professor
Rajesh Saturi[2], Assistant Professor
Anupiya Bharti[3], Meghana Avvari[4], Bhavana Battu[5]
Department of Computer Science and Technology,
Vignana Bharathi Institute of Technology, Aushapur, Ghakesar.

***Abstract* - For teachers, manually developing interesting and pertinent questions is a demanding and time-consuming endeavour. A useful yet difficult task in natural language processing is MCQ Generation System. Using a collection of options serving as distractions, this solution creates MCQ questions and answers. The aim is to create accurate and pertinent questions using textual information. The sentences are summarised using the BERT XL NET algorithm after the data has been analysed. Using the tags for the parts of speech, questions are constructed (POS Tagging). Teachers may generate multiple choice questions with the help of our MCQ Generator System, which also saves them time and effort. By concentrating on automating the process of creating test questions, this system bridges the gap between manpower and technology.**

***Keywords* -** *Natural Language Processing, BERT XL NET, Wordnet, PKE, Sentence Mapping.*

I. INTRODUCTION

As of now, creating quizzes and exam papers by hand takes a lot of effort from professors and teachers. Students also devote a lot of time to self-analysis. Also, students rely on their mentors for help with the self-analysis. So, we are working in this field of NLP, which is currently ripe for development. We intend to create a computer application system that will enable you to gauge your own performance and abolish any reliance on mentors. Here, students can enter the text of any source they used, and based on that, they will receive a set of questions and answers from which they can perform a self-analysis. Mentors create test questions and quizzes in a manner analogous to this.

Automated MCQ generation is the process of automatically creating questions from a variety of input forms, such as text, a structured database, or a knowledge base, that are syntactically sound, semantically right, and relevant. Online courses, creating objective/subjective questions, automated help systems, search engines, Chabot systems (e.g., for consumer engagement), and healthcare for mental health analysis are just a few examples of the numerous fields where MCQ can be used naturally. Despite its value, manually writing pertinent and meaningful questions is a difficult and time-consuming operation. For instance, it is tiresome for a teacher to manually design questions, obtain answers to those questions, and then evaluate answers while testing pupils' reading comprehension.

We can simplify the work of an instructor by creating this digital programme. Knowing the right questions to ask for the provided text input might save a lot of time. So, our goal is to create a system that can produce a variety of logical queries from the provided text input. Only humans can currently complete this at this time.

With the power of NLP, questions are automatically generated using the Automatic MCQ Generator. The system receives the text of any domain as input, which is then condensed using the BERT algorithm. A pre-trained model from Google called BERT (Bidirectional Encoder Representation from Transformers) uses deep learning to interpret natural language. Now, using the Python Keyword Extractor (PKE), the keywords are chosen from the condensed text, and a keyword is mapped appropriately to a phrase. A MCQ choice will include this keyword. Now, creating pertinent diversionary activities is

the major responsibility. The wordnet 2 method is used to produce distracters.

II. LITERATURE SURVEY

Automatic Question Paper Generation, according to Bloom's Taxonomy, by generating questions from text using Natural Language Processing [1], suggests questions from a passage provided as input using Natural Language Processing Library like NLTK and Spacy. The questions for each level of Bloom's Taxonomy are chosen at random to create the question paper.

Designing an Adaptive Question Bank and Question Paper Generation Management System [2], The entered question items are scored according to their level of difficulty, and the question paper can be adjusted to match the testing requirements, from easy to challenging. The evaluation module creates answer clues for the subjective questions and password-protected, expert-validated answer keys for the objective questions.

Automatic Question generation and answer assessment: A survey [3], A study of methods for creating questions and scoring tests automatically using textual and visual learning materials. This survey's goal is to compile the most recent methods for automatically creating questions and evaluating respondents' responses.

Automated Question Generator System Using NLP [4], The proposed system performs well when using paragraphs of text as input and the NLP libraries. It focuses on the analysis of a sentence's semantic and syntactic structure. One of the most important aspects of natural language processing that aids in finding the appropriate tag for a given text is part of speech tagging.

Automatic Multiple Choice Question Generation From Text: A Survey [5], Questions are created using articles in the database. The frequency of words in the text is counted using an NLP-based summarizer, and the key is chosen using pattern matching techniques. They have

employed the wordnet, pattern matching, domain ontology, and semantic analysis to generate the distractors.

III. EXISTING SYSTEM

In the current system, questions are created from text summaries that are provided as input. When text has been condensed, keywords are removed from it, and then the co-occurrence matrix of the words is found using the RAKE method. The word with the greatest score is then chosen as the sentence's keyword, and word-net is used to produce distractors for it.

The RAKE algorithm is the one utilised in the current system, and despite being precise and efficient, it has some limitations:

- The RAKE stop word list is insufficiently comprehensive; it would consider continuous long text as a phrase and produce extremely long phrases.
- Stop-word-filled expressions with many words could be overlooked. For instance, if "good" is on the stop word list, it can be missed if a company called "Good Day" is mentioned.

IV. PROPOSED SYSTEM

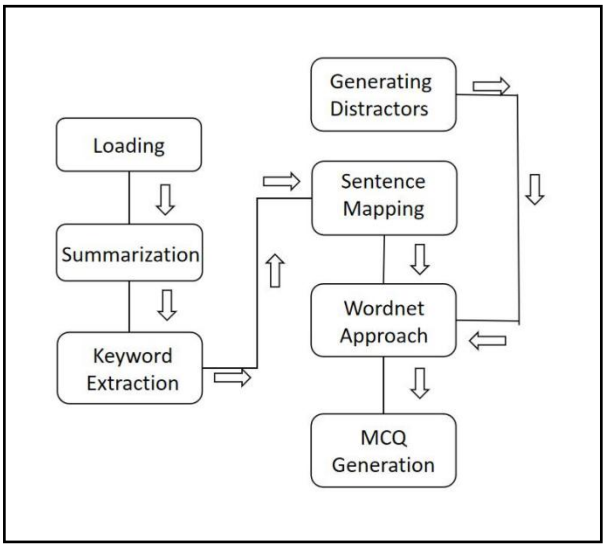
To create MCQs, various actions must be taken. The first phase is cleaning up a user-provided input text. Use the BERT XL NET summarizer to condense the cleaned-up text. The next phase involves extracting keywords from the condensed text using the PKE algorithm and then associating those keywords with the appropriate sentences. By determining the word sense of each word in the sentence, the Wordnet and Concept Net approach's final stage involves creating the distractor sentences. MCQs are created as a result.

The proposed method has the following benefits:

1. It minimizes the need for human intervention.
2. It saves money and time.
3. The questions and distractions generated are reasonably accurate.

4. Students who are studying for competitive exams can use this project, which can be employed by a variety of educational institutions.
5. The PKE algorithm is employed, and it can more precisely retrieve keywords that contain inner stop words than the alternative technique.

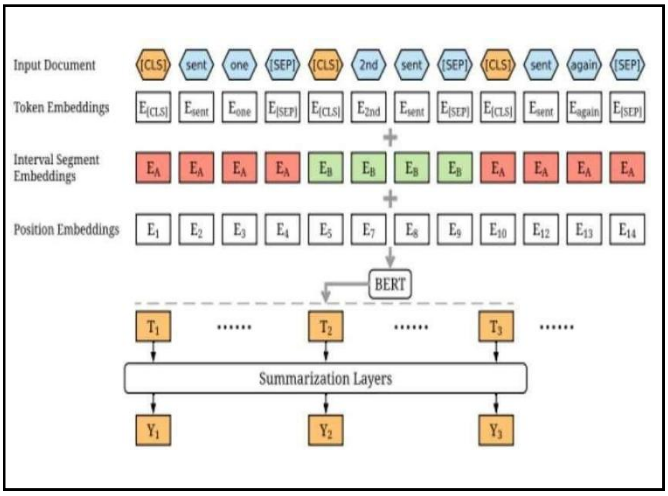
V. ARCHITECTURE



VI. METHODOLOGY

1. SUMMARIZING THE DATA

There are no questions that may be asked about any given sentence. Only sentences with a dubious fact can be used to create multiple-choice questions. Consequently, while creating MCQs automatically, sentence choice is quite important. Thus, the BERT Algorithm is employed to summarise the content. BERT (Bidirectional Encoder Representations from Transformers) is a neural network-based natural language processing (NLP) technique. It is designed to generate powerful contextual word representations by leveraging the bidirectional nature of transformers.It is a Google pre-trained open-sourced model. That makes it easier for computers to comprehend language a little more as humans do. The BERTSUM model, which is a refined version of BERT for extractive summarization, is used to summarise the input text. It displays the BERTSUM model's architecture.



In the BERTSUM model, a specific tokenization and embedding process is employed. At the beginning of each sentence, a [CLS] token is added, while a [SEP] token is inserted between every two sentences to serve as a separator. The [CLS] token captures the preceding sentence context. Each word in the sentence is tokenized using token embeddings. Notably, there is a distinction in segment embeddings. In BERTSUM, each sentence is assigned an embedding of Ea or Eb based on whether the sentence is even or odd. For example, if the sequence is [s1, s2, s3], the corresponding segment embeddings would be [Ea, Eb, Ea]. This way, all sentences are embedded and passed through subsequent layers.

BERTSUM then assigns scores to each sentence, reflecting the importance of that sentence in the overall document. Consequently, the sequence [s1, s2, s3] is associated with scores [score1, score2, score3]. The sentences with the highest scores are selected and reorganized to create a summary of the input text. The output generated by BERTSUM provides a concise and condensed representation of the original content.

2. KEYWORD EXTRACTION

The content is summarised, and then the sentence's keywords are chosen. The response to the query will be this keyword. As not every word in a sentence may function as a key, careful keyword selection is necessary. The Python library performs the keyword extraction. YAKE. A popular keyword extraction technique called Yet Another Keyword Extraction (YAKE) finds the most important words and phrases in a text by using a list of stopwords and phrase delimiters. There are primarily two parts to this method.

a. Preprocessing and candidate generation:

Space and special characters (line breaks, brackets, commas, and periods) are used as the delimiters to separate the sentences into terms. We choose the keyword's maximum length before it is formed. If we choose a maximum length of three, a sliding window is used to produce candidate phrases of one, two, and three grammes. Then, we eliminate sentences that contain punctuation. Moreover, stop words at the start and conclusion of phrases are eliminated.

b. Candidate scoring:

The text is represented by a graph created by this model. The most crucial lexical units (in this case, words) in the text are subsequently extracted using a graph-based ranking algorithm. In this version, the edges reflect the co-occurrence relation, which is determined by the distance between word occurrences, and the nodes are words of a specific part of speech (nouns and adjectives) (here a window of 2 words). The TextRank graph-based ranking algorithm's unweighted version ranks nodes.

3. SENTENCE MAPPING

The highest-scoring potential keywords are chosen as keywords once candidate keywords have been scored (where T is the number of keywords that has be extracted). The method will provide the top 3 keywords for the aforementioned example, which are PKE, keyword extraction (4.3.3), and numerous libraries. Following the selection of the keyword, each relevant sentence that contains the word from the condensed text is mapped for that keyword.

4. GENERATION OF DISTRACTORS

The most important phase in the creation of automated MCQs is the generation of distractors. The equality of the created distractors has a significant impact on how tough MCQs are. A good deterrent is one that closely resembles the key but is not the key. Thus, the Wordnet technique is employed to generate distractions.WordNet, developed at Princeton University, is a lexical database for the English language that serves as a component of the NLTK corpus. The words in the WordNet network are linked together by linguistic relationships. These linguistic relationships include meronym, holonym, hypernym, and hyponym. Synsets, which are collections of cognate synonyms made up of nouns, verbs, adjectives, and adverbs, are how WordNet organises and saves synonyms. Each word in a synset has the same meaning.

Each synset is essentially a collection of synonyms. There is a definition for every synset. Relations between various synsets are stored. The Lesk algorithm is based on the concept that words within a specific "neighborhood" or context in a given text often share a similar subject or topic.

A condensed version of the Lesk algorithm compares the terms in a word's neighbourhood with the meaning found in the dictionary. For instance, if the term "bat" appears in a phrase like "The bat flew into the jungle and settled on a tree," we immediately recognise that we are referring about the wingless mammal bat and not a cricket bat or baseball bat. Although we humans are skilled at it, algorithms struggle to tell one from another. The term for this is word sense disambiguation (WSD).

5. WORDNET

Developed at Princeton University, WordNet is a lexical database for the English language that serves as a component of the NLTK (Natural Language Toolkit) corpus. The words in the WordNet network are linked together by linguistic relationships. These linguistic relationships include meronym, holonym, hypernym, and hyponym. Synsets, which are collections of cognate synonyms made up of nouns, verbs, adjectives, and adverbs, are how WordNet organises and saves synonyms. Each word in a synset has the same meaning. Each synset is essentially a collection of synonyms. There is a definition for every synset. Relations between various synsets are stored. The underlying principle of the Lesk algorithm is based on the notion that words occurring within a particular "neighborhood" or context in a text tend to share a similar subject matter.A condensed version of the Lesk algorithm compares the terms in a word's neighbourhood with the meaning found in the dictionary. Wordnet is used to produce the appropriate distractors for each keyword. Hypernyms, hyponyms, and antonyms are examples of distracters that might confuse users. Following the creation of distractor words in a sentence that matches a keyword, the keyword is left blank and a question is created.

6. MCQ GENERATION

This stage produces a set of four possibilities or "distractors" and multiple choice questions with answers. The sentence that matches to each of the extracted keywords is used to frame the question. Also, as was already said, the appropriate collection of detractors is presented as a choice for the question beside the right answer.

VII. RESULTS

VIII. CONCLUSION AND FUTURE SCOPE

Multiple Choice Questions (MCQs) are successfully generated. Quality distractors are used to create effective queries. With the suggested solution, the issue of manually creating questions is resolved. The suggested system uses NLP to generate automated queries, reducing the need for human participation while saving time and money. And the distractor that is produced is passably effective. This approach assists students who are prepared for difficult exams as well as teachers with electronic assessments. Students can assess their problem-solving skills and topic comprehension by answering the questions. The creation of a test using Bloom's Taxonomy and including both multiple-choice and lengthy answer questions is the next phase of this system's development.

IX. REFERENCES

[1] Shivali Joshi, Parin Shah, Sahil Shah, “Automatic Question Paper Generation, according to Bloom’s Taxonomy, by generating questions from text using Natural Language Processing” *International Research Journal of Engineering and Technology.*

[2] Pankaj Dwivedi, R. Tapan Shankar, B. Meghana, H. Sushaini, B. R. Sudeep, M. R. Pooja “Designing an Adaptive Question Bank and Question Paper Generation Management System” *Advances in Intelligent Systems and Computing*

[3] Bidyut Das, Mukta Majumder, Santanu Phadikar, Arif Ahmed Sekh, “Automatic question generation and answer assessment: a survey” *Research and Practice in Technology Enhanced Learning*

[4] Priti, Gumaste Shreya, Joshi Srushtee, Khadpekar Shubhangi, Mali, “Automated Question Generator System Using NLP” *International Research Journal of Engineering and Technology*

[5] Dhawaleswar Rao, Ch Sujan Kumar Saha, “Automatic Multiple Choice Question Generation From Text: A Survey” *IEEE*