Automatic Question Generation from Handwritten Lecture Notes on KeyBERT-indexed T5-TrOCR Pipeline with Gemini Context Correction

Rommel John Ronduen[1], Jan Adrian Manzanero[2], Analyn Yumang[3]
School of Electrical, Electronics, and Computer Engineering
Mapua University
Manila, Philippines
[1]rjhronduen@mymail.mapua.edu.ph
[2]jacmanzanero@mymail.mapua.edu.ph
[3]anyumang@mapua.edu.ph

***Abstract*—In this research, a system capable of performing optical character recognition (OCR) on handwritten lecture notes and consequently using the extracted text for automatic question genertion (AQG) was conceived. The system utilized the Text-to-Text Transformer (T5) for AQG and the Transformer-based Optical Character Recognition model (TrOCR) for OCR. The base version T5 was fine-tuned using the SQuADv1.1 dataset while the pre-trained handwritten base version for TrOCR was used. The system was evaluated using the word error rate (WER) for OCR evaluation, while the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) and Bilingual Language Understanding Evaluation (BLEU) were used for AQG evaluation. The system achieved a WER of 0.40 and a question validity rate of 68%.**

***Index Terms*—Large Language Model, Optical Character Recognition, Automatic Question Generation, Handwritten Lecture Notes, Raspberry Pi**

I. Introduction

Handwritten lecture notes are still widely used in educational institutions. Since the advent of artificial intelligence (AI), education has become the primary focus of AI research. AI has been used to automate several processes, and one which is the generation of learning materials such as questions in the form of quizzes. Literature defines automatic question generation (AQG) as the process of generating questions from a given text.

Despite existing implementations of AQG, no attempt has been given for utilizing handwritten lecture notes as a source for AQG. There have been several approaches to AQG, such as rule-based, template-based, and neural-based, but none of these approaches have been applied to handwritten lecture notes as a direct source for AQG through image capturing and OCR. It is noted by (Arbaaeen) AQG is defined by the methods used to generate questions. The AQG system is composed of three main components: the context extraction and parsing, the question generation, and the question validation. AQG systems differ in the method of extracting information and also the type of information that is being extracted. In fact, it was noted by them that it is suggested that AQG implementations utilize other forms of media sources for context bases. For instance, (Gaur) conceived of an AQG system that utilizes programming source code as the context for question generation based on selected keywords from the code snippets. However, these code snippets were assumed to be transcribed as an input to the system and not from the handwritten medium. Another use case through (Ou) involved the use of recorded videos as a source for AQG that led to question-answer pairs through the Bidirection and Auto-Regressive Transformer (BART) model for the search of sentences in the text. They utilized an algorithm of indexing or searching through the extracted context for specific question generations, which will be explained further in the succeeding text. Despite this, the researchers utilized a recorded video as a source. On the other hand, (Moron) allowed for an implementation of AQG through the use of named entity recognition (NER) and semantic rules for question generation in aiding the learning of English that led to the generation of questions that primarily answered the questions of ”what” and ”who”. However, the AQG implementation relied on the manual input of the user in text boxes for the context. In total, AQG systems are capable of generating questions from a given context, but no system has been developed to generate questions from handwritten lecture notes. There exists a variety in the algorithms used for AQG, such as the use of the BART model for question generation through the use of a search algorithm and the use of NER and semantic rules for question generation. An example includes the use of treebanks for providing for inferences (Padilla). However, one such study by (Tsai) utilized the use of the Text-to-Text Transformer (T5) model for AQG and the evaluation utilizing the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) and the Bilingual Evaluation Understudy (BLEU) metrics. The Stanford Question Answering Dataset (SQuAD) was used for the fine-tuning of the T5 model that led to satisfactory similarity of model-generated questions to the ground truth questions with a ROUGE-L score of 0.613. However, they noted a limitation as the model had a BLEU score of 0.567 due to the processes of the T5 model mixing the syntax and form of the context used for AQG. Due to the capability of the T5 model to generate questions from a given context while being fine-tuned to a specific dataset, it was chosen as the model for AQG in this research. In bridging the gap for AQG from handwritten lecture notes, the Transformer-based Optical Character Recognition (TrOCR) model was used for the optical character recognition (OCR) of the handwritten lecture notes. Starting first with OCR, it is often associated with convolutional neural networks (CNN) and recurrent neural networks (RNN). As noted by (Manlises) the use of CNNs for OCR is often used for the detection of text in images and even objects such as that of the implementation for the detection of the different types of mushrooms (Caya). CNNs have also been used for recognizing text through the transformation of shorthand terminologies to English text (Vitug). In fact, it was shown by (Ligsay) that it was possible to recognize text in Baybayin (a Filipino lettering system) using CNNs. Another involved the identification of expiry dates on canned goods (Manlises). In terms of evaluation, these CNNs are often evaluated using confusion matrices. These confusion matrices are used to evaluate the performance of the OCR model in terms of the true positive, true negative, false positive, and false negative values (Villaverde). However, a transformer-based approach has been used for OCR, such as the TrOCR model (Li) which has been used in recognition of text from scanned receipts (Zhang) and even in the recognition of text from images of Arabic text (Mortadi). The model has been exemplary over the use of CNNs by its encoder-decoder framework with pre-trained weights for the recognition of text.

In lieu of the gap on AQG from utilizing handwritten lecture notes, the general objective of this research to develop a system of allowing for automatic question generation from handwritten lecture notes on KeyBERT-indexed T5-TrOCR pipeline with Gemini context correction. The specific objectives of this research to utilize a fine-tuned T5 model for AQG on SQuADv1.1 while using KeyBERT for indexing the context of the handwritten lecture notes; to utilize the base handwritten TrOCR model for the OCR of the handwritten lecture; to evaluate the system using the word error rate (WER) for OCR evaluation and the ROUGE and BLEU metrics for AQG evaluation; and to utilize a Raspberry Pi 5 within a constructed enclosure with proper illumination for the facilitation and procurement of the system processes, a web camera for image capture, and a touchscreen monitor for the display of the generated questions.

This research is mainly limited by the consideration of only single-column, diagram and equation free, and English handwritten lecture notes of no erasures on strictly letter-sized plain sheet paper. Moreover, the system is confined to the use of the T5-TrOCR pipeline utilizing the base versions while also utilizing the SQuADv1.1 dataset for the fine-tuning. It is also important to note that the system is limited to the facilitation of the processes in the hardware of the 8-gigabyte version of the Raspberry Pi 5. The use of a higher parameter count for the T5 and TrOCR models and the use of a different dataset for fine-tuning may allow for differing results. Moreover, the utilization of a different hardware setup may also lead to alternative, if not, better results in terms of processing time.

II. Materials and Methods

A. Hardware Development

1) System Block Diagram:

2) Experimental Setup:

B. Software Development

1) System Flowchart:

2) Model Fine-tuning:

C. Data Gathering

D. Testing and Evaluation

III. Results and Discussion

IV. Conclusion and Recommendations

Acknowledgment

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks …”. Instead, try “R. B. G. thanks…”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

References

Please number citations consecutively within brackets [**?**]. The sentence punctuation follows the bracket [**?**]. Refer simply to the reference number, as in [**?**]—do not use “Ref. [**?**]” or “reference [**?**]” except at the beginning of a sentence: “Reference [**?**] was the first …”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [**?**]. Papers that have been accepted for publication should be cited as “in press” [**?**]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [**?**].