Question and Assessment Generator: Deep Learning Approach for Customizable and Intelligent Assessment Creation

2024 IEEE International Conference for Women in Innovation, Technology & Entrepreneurship (ICWITE) | 979-8-3503-8328-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICWITE59797.2024.10503319

Snehal Rathi
Computer Department
Vishwakarma Institute of Information Technology
Pune, India
snehal.rathi@viit.ac.in

Vaishnavi Pophale
Computer Department
Vishwakarma Institute of Information Technology
Pune, India
vaishnavi.22010473@viit.ac.in

Prajwal Kutwal
Computer Department
Vishwakarma Institute of Information Technology
Pune, India
prajwal.22010845@viit.ac.in

Shivam Mishra
Computer Department
Vishwakarma Institute of Information Technology
Pune, India
shivam.22010678@viit.ac.in

Atharva Nadkarni
Computer Department
Vishwakarma Institute of Information Technology
Pune, India
atharva.22010194@viit.ac.in

***Abstract*— An essential tool used by educational institutions to evaluate student performance is the examination process. The type of exam questions would dictate the caliber of students that the colleges produced. For the instructors, creating the exam questions is difficult, time-consuming, and unpleasant when they are new to the subject. For this reason, having an intelligent system that assesses students' learning abilities is crucial. We have proposed an automated question paper generation process that is quick, efficient, secure, and randomized. The exam paper is created and the inspection is completed by the design procedure using an efficient algorithm that has a high success rate. It is powered by deep learning and natural language processing (NLP) methods, particularly by using transformer models. With ease, the system can process a wide range of input formats, such as PDFs, audio files, and video, and transform them into a format that is standardized for efficient analysis. We have also tried to explore how to integrate Bloom's Taxonomy learning methods into the questions. Field-specific algorithms provide even more flexibility by guaranteeing that the question-creation process is in line with the particularities of each field which is important for subjects like mathematics. To provide a thorough assessment experience, the generator allows the construction of a wide range of question kinds, such as Fill-in-the-blanks (FIBs), Multiple-Choice Questions (MCQs), and Abstract question generation along with Mathematical word problems. Teachers benefit additionally from an automatically created model answer paper from the system, which makes paper correction quick and reliable.**

Keywords - Natural Language Processing, Transformers, Deep Learning, Question and Assessment Generator.

I. INTRODUCTION

The paradigm shift in education that has occurred recently toward personalized and technologically enhanced learning calls for creative methods of assessment and evaluation. The system is intended to transform the traditional process of creating assessments by integrating data entered by users in a variety of formats, including PDFs, audio files, and video. This generator promises to provide educators with a dynamic tool for creating assessments that not only reflect the breadth of student understanding but also meet the unique needs of both teachers and students through the use of sophisticated algorithms and machine learning techniques.

The ability to seamlessly integrate various data formats is one of the main challenges in assessment design. This problem is addressed by our model, which offers a reliable method for transforming various formats into a common structure and establishing the groundwork for an exhaustive examination. Furthermore, by using machine learning algorithms that are in line with Bloom's Taxonomy, it is ensured that the questions that are generated cover a wide range of cognitive domains, making it possible to assess students' depth of comprehension, learning capacities, and critical thinking abilities. One notable feature that truly elevates the depth and complexity of the assessment creation process is the incorporation of Bloom's Taxonomy. Learning objectives and outcomes can be categorized using Bloom's Taxonomy, a hierarchical model of cognitive skill classification that is well-known in the field of education. Cognitive skills like remembering, understanding, applying, analyzing, evaluating, and creating are divided into levels by Bloom's Taxonomy. Through the integration of machine learning algorithms that are in line with Bloom's Taxonomy, the system surpasses basic assessments and makes it possible to differentiate evaluations according to the complexity of cognitive skills that are needed. This guarantees that tests cover a range of cognitive skills, from simple memory to complex critical thinking.

The generator uses subject-specific algorithms because it understands that various subjects are subtle in different ways. This guarantees that the process of creating questions is customized to the distinct educational goals linked with every subject. As a result, the algorithms governing assessments in mathematics are very different from those governing assessments in English, indicating a dedication to subject-specific subtleties in the assessment creation process. It is crucial to use Mathematical Word Problems (MWPs) to help

979-8-3503-8328-7/24/$31.00 ©2024 IEEE

students develop their literacy and numeracy abilities. MWPs have historically been created manually; however, authentic, varied, and configurable research and education can both greatly benefit from an efficient automated MWP generator. The generator facilitates the creation of multiple-choice questions (MCQs), fill-in-the-blanks (FIBs), and abstract question answers to increase the adaptability of assessments. This flexibility guarantees a thorough assessment that takes into account various learning preferences and encourages a more comprehensive comprehension of students' knowledge. Because of this flexibility, tests are meaningful to students with different learning styles, which promotes a more comprehensive evaluation of their comprehension.

In the context of education, it is crucial to guarantee the caliber of assessments. To maintain the integrity of the assessment process, the system includes a mechanism to reject questions that are not well-formed. Additionally, to help teachers grade assignments more quickly and consistently, the generator creates a model answer sheet which is automatically generated.

This study represents a major advancement in the development of intelligent and customized assessment tools, which have the potential to fundamentally alter the nature of education by giving teachers an effective tool for customizing exams to each student's specific requirements. The methodology, specifics of the implementation, and possible implications of the Question and Assessment Generator are covered in more detail in the following sections. Our paper has six sections containing the Literature Survey, the Proposed Work which consists of the subsections: Taking input from the user, Bloom's Taxonomy Theory, Dataset Information, Question Generation which consists of MCQ, FIB, Wh and Abstract questions, Mathematical Word Problems and Open-ended Questions. Then we have the Model Overview of the models mainly used followed by the Results section and the Conclusion along with the future scope section.

II. LITERATURE SURVEY

This is a comprehensive overview of previous works in the field of automatic question generation, offering insights into diverse methodologies and tools employed in this domain. Through an examination of key studies, this survey aims to lay the foundation for understanding the landscape of automatic question generation, ranging from language-specific approaches and theoretical frameworks to advanced Natural Language Processing (NLP) techniques and innovative neural network architectures. Bloom [1] classified taxonomies based on educational aims as complicated or specific. On how the questions must be formulated, he proposed six degrees of cognitive learning: Remembering, comprehending, applying, analyzing, evaluating, and producing. Rus and Graesser [2] separated questions into two categories: deep and shallow questions, which are similar to Bloom's. Here, deep questions involve a conceptual understanding of questions like why, how, and what-if from complex sentences or a passage with two or more sentences, while shallow questions concentrate on facts like who, when, etc. from simpler factual sentences. Creating easy-to-answer, superficial questions is our primary goal. Heilman and Smith's more recent paper [3] outlined several difficulties that may arise both during the question-asking and answer-giving phases. These presented difficulties with discourse, lexicon, and syntax. These difficulties highlight how hard it is to come up with questions that are both syntactically and semantically sound within the given input text. Neural network-based methods have been employed more recently to address question generation. To create questions, Serban et al. [4] worked with a triplet of subject, relation, and object. Questions were created using the relation.

A max-out pointer network was proposed by Zhao et al. in [5] to monitor word coverage. From paragraphs, it produced questions. To create the questions, a sophisticated deep-learning technique was employed. However, several evaluation metrics have been developed for sentences, but not specifically for questions. Kishore et al. in [6] mentioned that one of the earliest metrics for evaluation is the BLEU score. It creates a score by n-gram comparison with the original sentence.

Question generation can also be done through semi-automated methods. Human-generated templates are used in semi-automatic question generation, along with database queries to finish the question as mentioned by Rey et al. [7]. Sherlock which is another template-based question-and-answer generator that makes use of linked data; it can produce questions of different levels of difficulty according to Liu and Lin et al. [8]. On the other hand, creating a sizable collection of excellent questions with semi-automatic question-generation techniques can be laborious and time-consuming. Additionally, the questions that can be created are limited by the templates. As a result, creating a big dataset of questions is difficult according to Rus et al. [9]. Mitkov and Ha et al. [10] stated other automatic question generators require human-made rules for the model to adhere to. To establish the guidelines for transforming declarative sentences into interrogative questions, educators are enlisted according to Wang et al. [11]. Adamson et al. [12] proposed an extension to a state-of-the-art question generation system that allows it to produce deep, subjective questions suitable for group discussion.

To generate rules the educator must be knowledgeable about both language and the subject matter. Similar to the previously discussed template-based techniques, this rules-based approach may also require a lot of mental effort and time. Additionally, Transformers can do better for a much lower expense of training. Like the RNN approach, transformers have an encoder and a decoder. Transformers also incorporate beam search and bucketing mechanisms. Unlike RNNs, transformers adopt multiple attention heads without requiring any recurrence, though recurrence can be added. Transformers can train faster than RNNs because they are more parallelizable and perform well with both large and small datasets Vaswani et al. [13]. Rathi et al. proposed that by tailoring focused interventions to pupils' unique affective states, the accurate affective-states prediction might enhance their learning gains [14]. The goal of the automatic question generator is to produce fresh, naturally occurring, semantically correct, and syntactically coherent questions from the text, as suggested by Rathi et al. [15]. Thus, if we can adjust the complexity of created questions, we will immediately enhance their quality. One of the most challenging issues is ensuring that the finished question set has a range of difficulty and covers the full content.

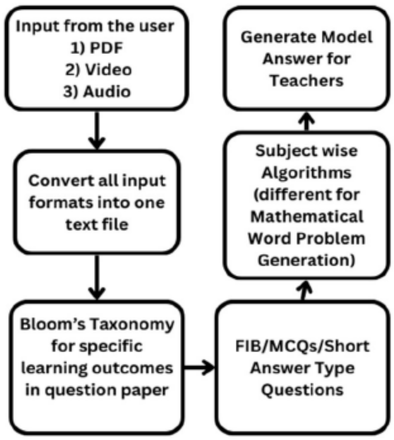


Fig. 1. Flow of the System

III. PROPOSED WORK

Several academics who work as instructors, tutors, and professors put a lot of effort into crafting their exam questions and quizzes. Likewise, students spend a great deal of time in self-reflection or self-calibration. Additionally, students turn to their mentors for help when conducting self-analysis. This has led to working in the domain of natural language processing (NLP), which still has a lot of space for growth. In addition, they have serious security concerns, and because the institutions lack teaching staff, producing papers is difficult. In conclusion, the system recommends an Automatic Question Generator that provides fast operations, data storage, and high security, we can see the workflow in Figure 1. The goal of the suggested system is to increase the efficacy of the current one. This method allows for the circumvention of every limitation.

A. TAKING INPUT FROM THE USER:

Our Proposed System first includes taking input from the user in different formats such as video, audio, or text. We then convert the input into a standard format such as a text file. To streamline the input acceptance and consolidation process for our innovative Question and Assessment Instrument Generator, we employ specialized models for audio, video, and PDF conversions into a unified text format. WhisperAI: is an automated speech recognition (ASR) system that was trained using web-sourced, 680,000 hours of multilingual, multitask supervised data. It is demonstrated to us that utilizing a dataset this size and variety improves resilience to background noise, accents, and technical language. It also makes multilingual transcription and translation possible, both in and out of English. To lay the groundwork for future research on robust speech processing and the development of practical applications, we are making our models and inference code publicly available. WhisperAI, our chosen machine learning model for speech recognition and transcription, handles audio files adeptly, swiftly transcribing spoken content into text. For video files, we utilize the moviepy module in Python to extract audio content, subsequently processed by WhisperAI for text conversion. Additionally, PyPDF2, a Python module, effectively processes PDF files, extracting textual information for further integration. By leveraging these models, we ensure a seamless transition from varied input formats like audio, video, and PDF—to a unified text format, enabling our system to analyze and generate standardized assessment instruments efficiently and accurately.

B. BLOOMS TAXONOMY THEORY:

Achieving Bloom's Taxonomy-aligned question generation involves leveraging various technologies, models, and methodologies. It involves a comprehensive understanding of Bloom's Taxonomy, which categorizes cognitive levels into six hierarchical tiers: Remembering, Understanding, Applying, Analyzing, Evaluating, and Creating. For Data Collection and Classification, we can Gather a diverse dataset of educational content aligned with different cognitive levels of Bloom's Taxonomy. Classify this dataset based on the cognitive difficulty levels. Then using the Generative pre-trained transformer, we can Preprocess the dataset, ensuring it is formatted and annotated according to Bloom's Taxonomy cognitive levels. Fine-tune the chosen model on the preprocessed dataset, emphasizing the different cognitive levels specified in Bloom's Taxonomy. We can run the model to understand and generate questions corresponding to each cognitive tier.

Then, we can utilize the fine-tuned model to generate questions by providing prompts specific to each cognitive level in Bloom's Taxonomy. For each level, we can have the following: (1) Remembering level: Prompt the model with tasks requiring memory recall or recognition. (2) Understanding level: Guide the model with queries that assess comprehension or interpretation. (3) Applying level: Provide scenarios where the model needs to apply knowledge to solve problems. (4) Analyzing, Evaluating, and Creating levels: Frame prompts challenging the model to analyze, evaluate, or synthesize information respectively. For Validation and Refinement, evaluate the generated questions for their alignment with the respective cognitive levels of Bloom's Taxonomy. For this proposed workflow, leveraging a Transformer- based model, such as the T5 (Text-to-Text Transfer Transformer), would be beneficial due to its versatility in natural language processing tasks, including question generation.

C. DATASET INFORMATION:

1) SQuAD1.1: A reading comprehension dataset called the Stanford Question Answering Dataset (SQuAD) is made up of questions crowdsourced from a collection of Wikipedia articles. Each question's response is either a passage of text, or a span, from the relevant reading passage, or it may be left unanswered. We used this dataset to train the model for FIB, WH- questions, and MCQ generator.

2) Race Dataset: With over 28,000 passages and close to 100,000 questions, RACE is a sizable reading comprehension dataset. The dataset comes from Chinese English exams that are given to students in middle school and high school. For machine comprehension, the dataset can be used as training and test sets. We used this Dataset for the generation of multiple options in the MCQ questions as well as to train t5 for abstractive summarization. Text Preprocessing includes removing punctuation marks from the text and converting the text into lowercase. The text cleaning process includes removing brackets '(', ')', '[', ']', removing multiple spaces, removing hyphen '-' as it is not recognized by spacy.

D. QUESTION GENERATION:

1) *MCQ, FIB, Wh, and Abstract questions using T5 and Race dataset:* MCQ generation involves a structured three-stage process aimed at delivering high-quality multiple-choice questions. The proposed method for MCQ generation involves a structured three-stage process aimed at delivering high-quality multiple-choice questions. Firstly, the Text Preprocessing phase initiates the process by preparing the input context provided by the user for question generation. Subsequently, in the Question-Answer Generation stage we employed a fine-tuned version of the T5 Transformer model, utilizing the SQuAD1.1 dataset containing a substantial corpus of question-answer pairs for training. In this stage we extract relevant information from the context to generate accurate questions and corresponding answers based on the input.

R denotes $number\ of\ sentences\ in\ the\ context \div number\ of\ questions\ required$

$$R<1$$

The context provided is too less so the system will search for the given topic in the Wikipedia module and populate the context from it and use it for context generation. Topics are identified using Spacy and then the content for topics from Wikipedia.

$$R >= 1$$

The context provided is appropriate and the given context will be used for question generation. Firstly, the Text Preprocessing phase initiates the process by preparing the input context provided by the user for question generation. Subsequently, the Question-Answer Generation stage employs a fine-tuned version of the T5 Transformer model, utilizing the SQuAD1.1 dataset containing a substantial corpus of question-answer pairs for training. This stage extracts relevant information from the context to generate accurate questions and corresponding answers based on the input. From the above stage Fill in the Blanks as well as factual questions can be generated. For the MCQ questions below stage is required for the options generating. The Distractor Generation phase enhances the MCQs by introducing plausible distractors to the questions. For this, the RACE dataset, a comprehensive reading comprehension dataset sourced from English examinations in China, is utilized. This dataset, serving as both training and test sets for machine comprehension, provides contextual information to create meaningful distractors. We can see the distractor vs BLEU Score in Table 2. By leveraging the question, answer, and context inputs, the system generates three distractors separated by the ¡sep¿ token. Additionally, to augment the quality of the distractors, we employed sense2vec, a resource containing word vectors that capture relationships among words, to extract three most semantically similar words to the answer. The recommended and experimented model for this process is the t5-small, with a learning rate set at 0.001, ensuring optimal performance in MCQ generation with improved accuracy and relevance. We can see the questions generated in Table 1.

2) *Mathematical Questions Generation:* To generate Mathematical problems, our method is based on the one proposed by Wang and SU et al. [21] and we have exclusively concentrated on linear equation issues. Priya, for instance, can paint a fence in five hours. If three people paint the same fence at the same pace as Priya, how long will it take them to finish? It concerns the use of dimensional units in mathematics and language. It entails carrying out the creation process and verifying the created problems' legitimacy (by comparing them to real textbook problems, that is). The narrative generator and the equation generator are two components. The set of known values and the set of unknown values are contained in the syntax. The set of equations is later included in the semantics section, after which the accuracy is verified. Create an equation and give each of the equation's quantities a dimensional unit in the equation generation section. Generation of Equations There are four steps in the procedure. There are guidelines followed in the wording. Next, we create a random equation with two sides that each contain a single variable belonging to the same dimensional unit. Next, we choose a variable within the equation and replace it with two new variables based on the instantiated rules of the dimensional units. Finally, we stop when the desired level of arithmetic operations is reached. The created equation is first transformed into a binary expression, and it is then represented using the postfix notation. The related variables provide the dimensional units of the intermediary nodes. We iterate through every Atomic Expression Tree in the provided tree, creating a sub-story for everyone on our own. Two child nodes, representing the variables, and a parent node, representing the operator, make up the synthesized equation. We also construct a substory. An extra duty in creating a backstory. All of the leaf nodes—aside from one that is left up to the student's response—have values assigned to them. Specifically, every empty space in a template must correspond to the keywords that are allocated to each of the corresponding nodes in an expression. The templates (where + indicates instantiation by inserting keywords and values into the slots), and how to join the substories together to create a single story for our running example. Using the template, generate the expression's sub-story: Priya has been running steadily for 80 minutes, while the other has been running for hours.

TABLE I

RESULTS OF ABSTRACT QUESTION GENERATION

Text	Generated Question
Life is a journey filled with experiences and lessons. Challenges shape character, while successes bring joy. Embracing both, one learns resilience and gratitude. Each day is an opportunity for growth, fostering connections, and creating memories. In the tapestry of existence, simplicity becomes the thread that weaves a meaningful and fulfilling life.	What is the author's purpose in writing the passage?
Friendship is a precious bond that enriches life. It brings joy, support, and understanding. Shared laughter lightens burdens, and shared sorrows lessen pain. True friends stand by each other through thick and thin. In their company, moments become memories, creating a tapestry of warmth and connection that lasts a lifetime.	What is the best title for the passage?

TABLE II

DISTRACTOR WITH RESPECTIVE BLEU SCORE

Distractor Number	BLEU score
1	46.8
2	33.8
3	34.47

2) Open Ended Questions: By analyzing the input text, the system formulates open-ended questions from the abstractive answers that delve into the details and nuances of the given information. The question aims to assess the student's understanding and interpretation of the content, fostering a comprehensive evaluation.

For open-ended question generation, the system employs abstractive summarization to grasp the topic's essence. Questions are then formulated based on the user's interpretation. Additionally, a model answer is generated, providing a reference point for evaluating diverse responses. This process promotes critical thinking and allows for subjective responses, enhancing the assessment's depth.

IV. MODEL OVERVIEW

There are different models used for different purposes which are described below. We can see the models, their process, and the output in the model architecture diagram in Figure 2.

A. Sense2vec:

Sense2vec used here, is a neural network model that generates vector space representations of words from large corpora. It is an extension of the infamous word2vec algorithm. Sense2vec creates embeddings for "senses" rather than tokens of words. A sense is a word combined with a label i.e. the information that represents the context in which the word is used. This label can be a POS Tag, Polarity, Entity Name, Dependency Tag, etc.

B. T5 Overview:

The Transformer model, particularly the T5 (Text-To-Text Transfer Transformer), stands out as a significant advancement in the realm of natural language processing (NLP). Unlike its predecessors, T5 is fundamentally structured as a simple encoder-decoder model. It introduces a groundbreaking concept of a "unified framework," transforming every language-related problem into a text-to-text format. This model, developed collaboratively by Google and Facebook, represents the latest evolution in the transformer series.

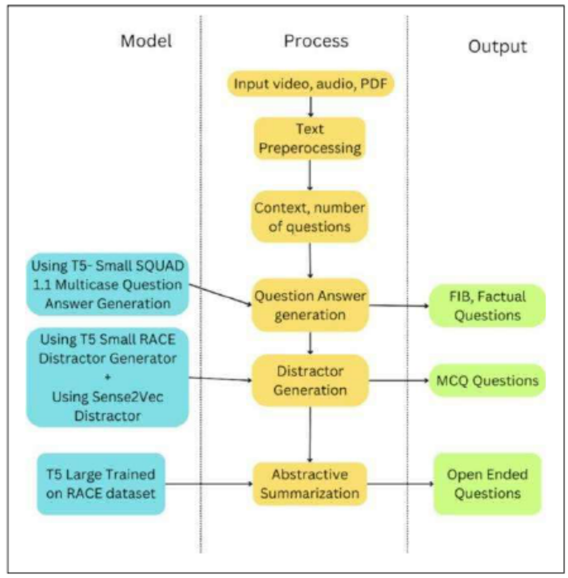


Fig 2. Model Architecture

One of T5's distinctive features is its inherent "text-to-text" nature. In contrast to conventional transformers that process natural language data after converting it into numerical embeddings, T5 operates exclusively with text inputs and outputs. This unique characteristic allows the model to seamlessly adapt to various NLP tasks without necessitating adjustments to hyperparameters or loss functions.

The term" unified" in T5's context implies its capability to concurrently perform multiple natural language generation (NLG) tasks. Unlike other transformers such as BERT and GPT2, T5 doesn't require separate output layers for different tasks. Even for regression tasks, the model can generate output in the form of a string of numbers. The training regimen involves a multi-task mixture of unsupervised and supervised tasks, with each task converted into a text-to-text format. The unsupervised training leverages the extensive Colossal Clean Crawled Corpus (C4) dataset, comprising 750 gigabytes of data specifically curated for T5, while supervised training utilizes well-known datasets tailored to respective tasks. As a result of its innovative design and training methodology, T5 has demonstrated superior performance across a spectrum of NLP tasks, surpassing several state-of-the-art architectures in approximately twenty different domains, including text summarization, question answering, and sentiment analysis. The input format for T5 follows a distinctive pattern, involving the addition of a special prefix to the standard input text. This prefix serves as an indicator of the task for which the model is expected to produce output. For example, in the context of translating a sentence from English to German, the input would appear as follows: "Translate English to German: That is good. "Here," translate English to German" represents the task prefix. These prefixes are selected during the fine-tuning process. Examples of other prefixes for pretrained tasks include" summarize" for text summarization, "cola" for assessing grammatical correctness, and "stab" for evaluating the similarity of two different sentences.

IV.DATASET USED FOR FINE-TUNING

The model is fine-tuned using the RACE dataset, which consists of passages and questions designed to assess reading comprehension. This dataset enables the system to understand and generate questions that align with the complexity and diversity found in real-world educational content. Basic parameters and results for training the T5 model are with a learning rate of 0.0001, batch size of 16, seed to be 42, gradient Accumulation Steps to be 16 and number of Epochs to be 7.

IV. TRAINING RESULTS

The model undergoes training with the specified parameters, achieving results that demonstrate its effectiveness in question generation tasks. We can see the training and validation loss vs Epoch graph in Figure 3. The Summary of the questions generated in the form of a graph can be seen in Figure 6.

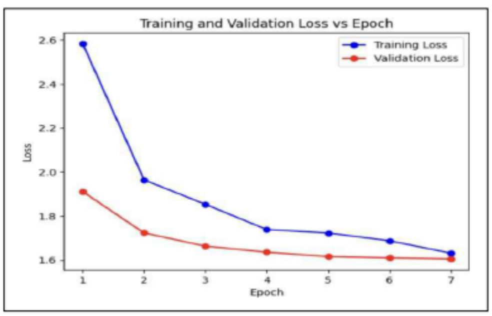


Fig 3. Training and Validation Loss VS Epoch

A. TEXT OR TEXT FILE TO PDF FOR QUESTION PAPER:

To create a question paper of the available text file, we can convert the text file to pdf format with proper header and footer input taken by the teachers containing the name of the school and name of the examination, etc. For this, we can use the fpdf library from Python.

Approach:

1) Import the class FPDF from module fpdf
2) Add a page
3) Set the font
4) Insert a cell and provide the text
5) extract the data from a text file using file handling.
6) Save the pdf with ".pdf" extension

B. MODEL ANSWER PAPER GENERATION:

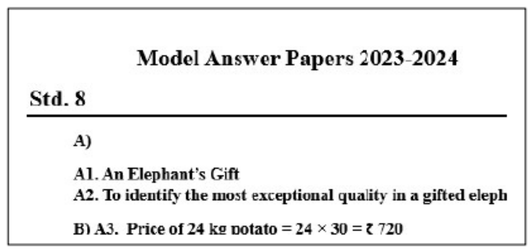
Generating model answers for teachers holds significant importance in the educational assessment process. It offers educators invaluable support by providing a standardized reference point for evaluating student responses. For that, we can use T5 fine-tuned on DuoRC for Generative Question Answering by just prepending the question to the context. The DuoRC dataset is an English language dataset of questions and answers gathered from crowdsourced AMT workers on Wikipedia and IMDb movie plots. We can see the results in Table 2.

TABLE II

RESULTS OF MODELS WITH DATASETS

Model	SelfRC	ParaphraseRC	SQUAD
T5-BASE-FINETUNED	F1:49.00 EM: 31.38	F1: 28.75 EM: 15.18	F1:63.28 EM: 37.24
BERT-BASE-FINETUNED	F1:47.18 EM: 30.76	F1:21.20 EM: 12.62	F1:77.19 EM: 57.81

Natural Language Toolkit in Python provides various modules and algorithms for symbolic and statistical natural language processing. It can be used to perform semantic analysis and syntax checking on generated questions. This involves parsing the questions to ensure they adhere to grammatical rules, have clear structure, and make semantic sense. And remove ill-formed Questions. We can see the Question Paper and Model answer paper generated in pdf format in Figures 4 and 5.



Model Answer Papers 2023-2024

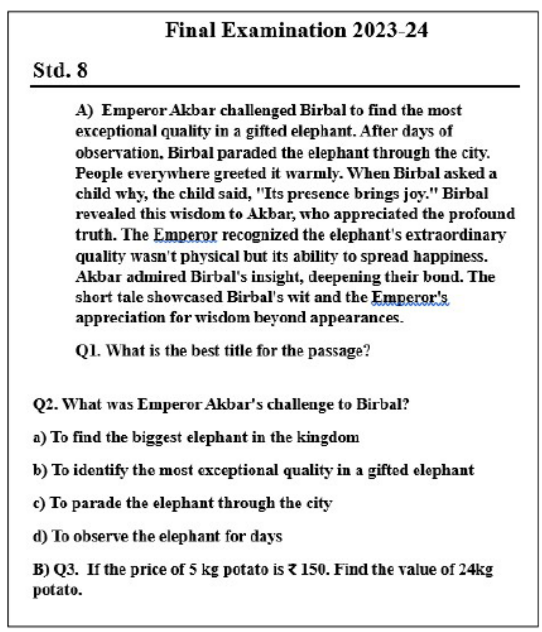
Std. 8

A)

A1. An Elephant's Gift
A2. To identify the most exceptional quality in a gifted eleph

B) A3. Price of 24 kg potato = 24 × 30 = ₹ 720

Fig. 4. Model Answer paper pdf



Final Examination 2023-24

Std. 8

A) Emperor Akbar challenged Birbal to find the most exceptional quality in a gifted elephant. After days of observation, Birbal paraded the elephant through the city. People everywhere greeted it warmly. When Birbal asked a child why, the child said, "Its presence brings joy." Birbal revealed this wisdom to Akbar, who appreciated the profound truth. The Emperor recognized the elephant's extraordinary quality wasn't physical but its ability to spread happiness. Akbar admired Birbal's insight, deepening their bond. The short tale showcased Birbal's wit and the Emperor's appreciation for wisdom beyond appearances.

Q1. What is the best title for the passage?

Q2. What was Emperor Akbar's challenge to Birbal?

a) To find the biggest elephant in the kingdom

b) To identify the most exceptional quality in a gifted elephant

c) To parade the elephant through the city

d) To observe the elephant for days

B) Q3. If the price of 5 kg potato is ₹ 150. Find the value of 24kg potato.

Fig. 5. Question paper pdf generated

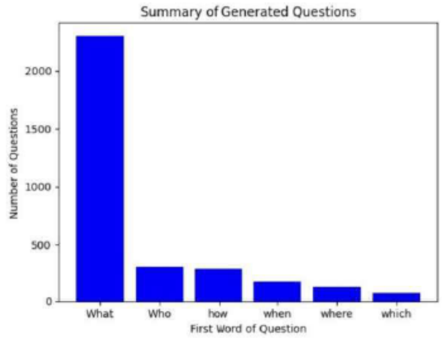


Fig. 6. Summary of questions generated

V. CONCLUSION AND FUTURE SCOPE

The proposed system can be used by teachers to generate a question paper based on the lecture to quickly understand how well the students have perceived the topic and also by the students themselves to understand their level of preparedness for the given topic. Additionally, we can facilitate the creation of a varied collection of excellent questions spanning disciplines by enabling educators to exchange and contribute to a collaborative platform. Additionally, we may improve the system such that it can automatically produce thorough justifications for responses, which will help students comprehend the logic behind the right answers.

REFERENCES

[1] Bloom, B. S. (Ed.). (1956). Taxonomy of Educational Objectives: The Classification of Educational Goals. New York: Longmans, Green.

[2] Rus, V. and Graesser, A.C. (2009). Workshop Report: The Question Generation Task and Evaluation Challenge, Institute for Intelligent Systems, Memphis, TN, ISBN: 978-0-615-27428-7Nan Duan, Duyu Tang, Peng Chen, Ming Zhou, Proceedings of the (2017) Conference on Empirical Methods in Natural Language Processing, pages 866–874.

[3] Heilman, M., and Smith, N. A. 2010. Good question! statistical ranking for question generation. 609–617. Los Angeles, California: Association for Computational Linguistics.

[4] Serban, Iulian & García-Durán, Alberto & Gulcehre, Caglar & Ahn, Sungjin & Chandar, Sarath & Courville, Aaron & Bengio, Y. (2016). Generating Factoid Questions with Recurrent Neural Networks: The 30M Factoid Question-Answer Corpus. 588-598. 10.18653/v1/P16-1056.

[5] Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. (2018). Paragraph-level neural question generation with max-out pointer and gated self-attention networks.

[6] Kishore Papineni, Salim Roukos, Todd Ward, and WeiJing Zhu. (2002). Bleu: a method for automatic evaluation of machine translation. In ACL, pages 311– 318. ACL.

[7] Rey, G. A.; Celino, I.; Alexopoulos, P.; ´ Damljanovic, D.; Damova, M.; Li, N.; and Devedzic, V. 2012. Semi-automatic generation of quizzes and learning artifacts from linked data.

[8] Liu, D., and Lin, C. 2014. Sherlock: a semi-automatic quiz generation system using linked data. In International Semantic Web Conference (Posters & Demos), 9–12. Citeseer.

[9] Rus, V.; Wyse, B.; Piwek, P.; Lintean, M.; Stoyanchev, S.; and Moldovan, C. 2010. The first question generation shared task evaluation challenge.

[10] Mitkov, R., and Ha, L. A. 2003. Computer-aided generation of multiple-choice tests. In Proceedings of the HLT-NAACL 03 workshop on Building educational applications using natural language processing, 17–22.

[11] Wang, W.; Hao, T.; and Liu, W. 2007. Automatic question generation for learning evaluation in medicine. In International conference on web-based learning, 242–251. Springer.

[12] Adamson, D.; Bhartiya, D.; Gujral, B.; Kedia, R.; Singh, A.; and Rose, C. P. 2013. Automatically generating discussion questions. In AIED.

[13] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. CoRR abs/1706.03762.

[14] Kurdi, G., Leo, J., Parsia, B., et al. A Systematic Review of Automatic Question Generation for Educational Purposes. Int J Artif Intell Educ 30, 121–204 (2020).

[15] Rathi, S.R. and Deshpande, Y.D. (2022), "Course complexity in engineering education using E-learner's affective-state prediction", Kybernetes, Vol. ahead-of-print No. ahead-of-print. https://doi.org/10.1108/K-09-2021-0806

[16] S. Rathi, P. Wawage and A. Kulkarni, "Automatic Question Generation from Textual data using NLP techniques," *2023 International Conference on Emerging Smart Computing and Informatics (ESCI)*, Pune, India, 2023, pp. 1-7, doi: 10.1109/ESCI56872.2023.10100278.

[17] Zichao Wang, Andrew Lan, and Richard Baraniuk. 2021. Math Word Problem Generation with Mathematical Consistency and Problem Context Constraints. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 5986–5999, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.