ProtonAi LLM based approach for question-answer automation

B. Ravinder Reddy
Assistant Professor
Dept. of CSE
Anurag University
Hyderabad, India
ravinderreddycse@anurag.edu.in

Gajawada Vishwas
Computer Science and Engineering
Anurag University
Hyderabad, India
gajawadavishwas@gmail.com

Racharla Sai Chaitanya
Computer Science and Engineering
Anurag University
Hyderabad, India
racharlasaichaitanya@gmail.com

Molugu Abhinav
Computer Science and Engineering
Anurag University
Hyderabad, India
abhinavmolugu659@gmail.com

Abstract-

This work presents a question answering system focused on extracting knowledge from a single textbook. We leverage Langchain, an open-source framework, for data handling and integration with large language models (LLMs) like OpenAI's GPT-3. ChromaDB, a vector database, efficiently stores and retrieves relevant passages. This combination allows for a focused Q&A system that leverages the LLM's capabilities while ensuring answers remain grounded within the textbook. We evaluate the system's efficacy through comprehensive benchmarking, demonstrating competitive performance in accuracy, response coherence, and contextual relevance. Furthermore, we explore real-world deployment scenarios to assess user-friendliness. This research contributes to the development of conversational AI by presenting a sophisticated, textbook-focused Q&A model. The findings have implications for applications like virtual assistants and educational platforms, where context-aware interactions are crucial. This work lays a strong groundwork for future improvements in conversational AI technologies.

Keywords - **LLM, Generative AI, GPT-3.5 Architecture, Educational Platform, Virtual Assistance, Chat-with-your-data.**

I. INTRODUCTION

Exploring the realms of intelligence development, technology plays a pivotal role in generating questions and answers. In today's interconnected world, where information is readily available on the internet, technology utilizes user input to create questions and answers based on book knowledge.

Embarking on This research explores Large Language Models (LLMs) in conjunction with Langchain libraries and the influential Generative Pretrained Transformer (GPT). The primary objective is to elucidate complex systems and their potential in question and answer generation.

Langchain serves as a crucial tool in this endeavor. It offers a structured framework specifically designed for interacting with LLMs, enabling us to leverage their capabilities for question and answer tasks. By utilizing Langchain's functionalities, we aim to simplify the process of crafting effective and informative questions that can elicit insightful and comprehensive answers from LLMs.

Subsequently, this paper outlines the journey from the collection of the data, pre-processing the data where the Langchain's libraries utmost efficiency is used to implement deduplication and get error free data. This enhances the models ability to generate a response in a more accurate and helpful manner.

Along with the usage of Langchain as it cannot become an AI on its own we combined it with the LLm's which will help the it to get more accurate and work efficiently. The chromadb vectore-stores part is used to store the data of the text-book knowledge and retrieve the answers based on the user query.

The data is store in the format of vectors which the AI can actually understand the RAG(Retrieval Augmented generation) plays a major role in this which is the efficient manner to retrieve the data even from the images as its summaries.

RAG works by combining the strengths of LLMs with information retrieval techniques. When a user poses a question or provides a prompt, the system first taps into an external knowledge base, which could be a vast collection of text and code, a specific database, or even web documents. This retrieval component identifies information relevant to the user's input.

In essence, this research paper navigates the fascinating intersection of language, technology, and innovation. It extends an invitation to explore the uncharted realms of AI, where questions are not merely answered but crafted through an artistic blend that blurs the boundaries between human and machine interactions.

II. LITERATURE REVIEW

Matt Dunn, Levent Sagun, Mike Higgins, and Volkan Cirik [1] introduced the SearchQA dataset for machine understanding and question answering. Unlike previous datasets, it incorporates Jeopardy! question-answer pairs extended with text snippets from Google, resulting in over 140,000 pairs with an average of 49.6 snippets each. The dataset, including metadata like snippet URLs, is deemed valuable for question-answering research. Human-machine evaluation reveals a notable performance gap, positioning SearchQA as a benchmark in the field. The dataset and evaluation code are publicly accessible.

Dubey Harish [2] presents an Automatic Question Paper Generation System using Python and fuzzy logic. The survey explores related works, such as adaptive question bank systems, fuzzy logic models, genetic algorithms, and Android applications for paper generation. The proposed system excels in reliability, deduplication, and user-friendliness, contributing significantly to automated question paper generation.

Mandar Joshi's [3] method utilizes the TriviaQA dataset, known for its reading comprehension challenges with over 650,000 question-answer triplets. Featuring compound questions, keywords, and inter-sentence reasoning needs, TriviaQA includes 95,000-word pairs, surpassing human performance with basic algorithms. The dataset, sourced from Wikipedia and the internet, serves as a rigorous test bed for reading comprehension models, presenting challenges beyond other big data sources.

Jonathan Berlant [4] proposes a novel approach to understanding words without complex rules, learning through questions and answers. This method, particularly effective for large datasets, outperforms other data methods, showcasing progress in enabling machines to comprehend language at scale.

Pascale Fung and Andrea Madotto [5] from Hong Kong University of Science and Technology address the issue of "illusion" in computer-generated language, where systems produce incorrect information. Investigating its occurrences in various contexts, the survey emphasizes the need to ensure reliable and non-harmful text on computers, especially in sensitive areas like medical applications.

Jared Kaplan and Benjamin Mann [6] provide a survey on recent advancements in natural language processing (NLP), specifically using large language models like GPT-3 with 175 billion parameters. Unlike traditional methods, GPT-3 demonstrates significant performance improvements across diverse tasks without task-specific optimization. The discussion highlights the challenges of needing large-scale domain data and the potential social implications of progress, addressing the limitations of current NLP techniques.

Rohan Kumar's [7] method tackles the challenge of long-term knowledge base rarity in open-source question answering. It proposes an automated approach to generate customized QA documentation for competitive products, using the Wikipedia knowledge graph to interpret content as a comprehensive academic record. The study evaluates GPT-3's performance on newly created long-term QA documents, emphasizing the potential for further research to enhance QA dataset generation and improve LLM performance.

Khushbu Khandait[8] introduces a novel system that self-generates questions based on students' thoughts, enhancing learning by asking effective questions and personalizing the process. The system, leveraging strategies like Transformers, proves to be a valuable tool for both teachers and students, making learning enjoyable and personalized.

Nithya M and Sanjeev Pranesh B[9] present a participatory approach highlighting how automatic question generation (AQG) aids teachers in designing effective questions. The use of techniques such as genetic algorithms, modified question banks, and fuzzy logic, along with artificial intelligence, particularly natural language processing, is emphasized. AQG accelerates the question creation process, improves accuracy, and reduces the workload for students.

Puneeth Thotad's [10] approach includes a question generator that uses natural language processing (NLP), a tool for quickly generating questions with many options from text. The system makes measuring comprehension easier by helping teachers and students measure comprehension. Leveraging Python programs with NLP libraries such as SpaCy and NLTK, the tool can process text, extract important data, and create queries. The research presented a variety of methods, including formal structure, keyword extraction, and Bloom's classification based on question design. The planning process will involve a series of algorithms that use NLP to analyze the input and generate different queries. The architecture process includes tools to provide teachers and students with good access and problem solving. Quizzes and performance tests show success in multiple choice, fill-in-the-blank, and Boolean-type questions from input. The system was designed to reduce manual work when creating tests and provide a foundation for future expansions, such as clarifying questions and analysis of answers.

Hala Abdel-Galil [11] introduces the Automatic Question Generation Model (AQGM) to facilitate exam creation in online education. AQGM uses a user-friendly GUI system, employing deep learning methods like segment-to-segment coupling with encoder-decoder, and achieved a notable BLEU4 score of 11.3 using the SQuAD dataset. The study emphasizes the importance of effective questioning in learning, and AQGM's interactive interface allows users to easily design tailored questions. The article concludes with future plans, including adding more problem types and advancing from auditing to improving performance.

Shivani G. Aithal [12] addresses limitations in question-answering systems (QAS) and proposes a solution called "similar questions." The method aims to enhance QAS performance by comparing questions to those generated from statements, identifying unanswered and irrelevant queries. The approach improves the system's focus on answerable questions, avoiding inaccuracies in responses. The article introduces an application for generating questions and answers from text, offering versatility. The study concludes with the author's declaration of data availability, absence of conflicts of interest, and adherence to an open-access license.

Rohan Bhirangi's [13] research focuses on transitioning from traditional writing to survey-based methods for schools, addressing biases and competition issues related to book-study information. The proposed automated system employs a role-based hierarchy and a mixed randomization algorithm to enhance efficiency, reduce bias, and improve test security. Tejas Chakankar's [14] research explores search engine-based learning, employing various methods like predetermined rules and deep neural networks to generate data-driven questions. Ongoing efforts focus on addressing challenges, such as appropriate problem selection and handling multiple correct answers, to enhance these systems and explore new applications.

Aanchal Jawere's approach [15] aims to simplify test creation for teachers through automated systems like AUTOQUEST. Leveraging natural language processing, this method accelerates the question generation process, making it quick, versatile, and unbiased, ultimately saving teachers time and improving the overall testing process.

Bidyut Das [16] discusses online learning methods, emphasizing reading, viewing images or videos, and listening to audio. Recognizing the need for questions and assessments to evaluate learning, Das recommends using automated systems for question creation and answer evaluation. The article reviews existing research, summarizes data, and highlights the growth of online learning, emphasizing the role of automation tools in facilitating the process.Abishek B. Rao's [17] work aims to enhance question-answer systems (QAS) in dealing with vast amounts of data. To address potential errors due to a lack of understanding, Rao proposes a pre-filtering step. Before reaching the QAS, questions are compared to possible ones in the description and scored, improving performance by prioritizing answerable questions. The study showcases creating question-answer pairs from articles, spanning QAS history, from early techniques to recent developments like BERT, emphasizing the role of natural language processing (NLP) in machine learning.

Safnah Ali's model [18] explores the application of Generative Adversarial Networks (GANs) in social media, particularly with ethical concerns related to technologies like deepfakes. The article suggests teaching intellectual skills to secondary school students through a Learning Path (LT) focused on GANs, machine-generated news, and ethical implications. Online workshops with 72 students demonstrated improved understanding of designs, applications, and critical thinking, highlighting the importance of AI education. Dr. Manoj Kumar's [19] research delves into text generation using computer models like recurrent neural networks (RNN), long short-term memory networks (LSTM), and gated recurrent units (GRU). The study tests these models on the Cornell University Film Archive, with a focus on chatbot history, including old techniques and modern technologies like bi-directional LSTM. Dr. Kumar compares the GRU and LSTM-based models, suggesting avenues for further text generation improvement.

Philipp Hacker [20] addresses the need for regulations like ChatGPT and GPT-4 to handle challenges posed by Large-Scale Artificial Intelligence (LGAIM) in the European Union and globally. Proposing a three-phase plan, Hacker suggests a basic model, additional rules for high-risk situations, and collaboration in the AI process. The report advocates for amendments to existing laws, such as the EU Artificial Intelligence Act and the Digital Services Act, to address specific LGAIM challenges, ensuring responsible use.

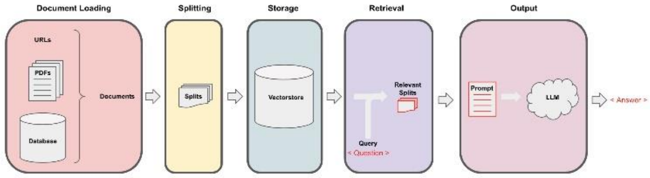
Vahid Ashrafimoghari [21] explores the use of smart computers, particularly in generating content for GMAT exam preparation. Testing GPT-4 Turbo against seven different computers, researchers found promising results, surpassing human performance. The study emphasizes the need for careful and balanced use of such services in education to ensure meaningful impact.

Deep Ganguli and Danny Hernandez [22] delve into the challenges of large-scale AI designs, noting both predictable "scaling laws" and unpredictable behaviors. The authors advocate for improving and responsibly managing these models by fostering collaborations between private companies and academic research, developing performance-checking tools, and testing new ways to maintain standards for the benefit of all.

Stefan Feuerriegel's [23] approach shows us how generative AI (a type of intelligence that produces content indistinguishable from human labor) can transform many businesses. The authors discuss examples such as Dall-E 2 and GPT-4, which demonstrate the potential for both artistic and practical use of this technology. They address challenges, including false positives and biases, and provide research methods for the business and knowledge engineering communities to address these issues and leverage intellectual intelligence capabilities.

III. PROPOSED METHOD

A. *Loading and Splitting PDF documents:* The code begins by loading several PDF documents related to Python using a library called PyPDFLoader. This library allows the program to access and extract text from the PDF files.



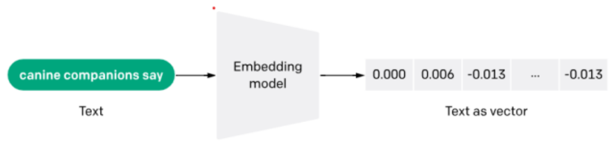
Next, the code splits the extracted text from each page into smaller, manageable chunks. It explores different splitting strategies using the langchain library:

Character split: This method breaks down the text character by character, offering fine-grained control over the chunk size and overlap between consecutive chunks.

Recursive character split: Similar to the character split, but it allows for creating nested chunks within the main ones, providing a more hierarchical view of the text.

Sentence split: This strategy leverages punctuation marks, particularly "/n/n" and "/n", to divide the text into individual sentences, preserving their integrity.

B. *Creating Text Embeddings:*



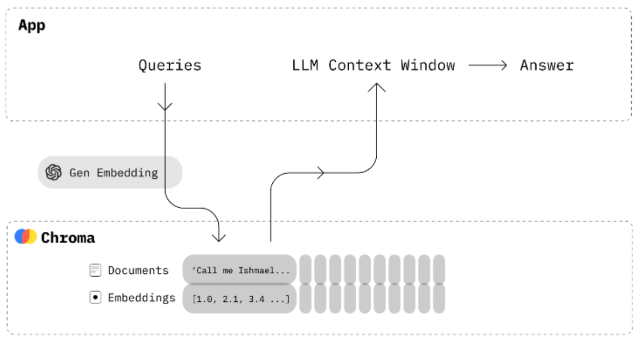
Text undergoes embedding using a model trained on extensive text, acquiring the ability to represent words as numerical vectors that capture meaning and context. This model outputs a vector representation for the entire text, converting it into a numerical format usable by machines in tasks like translation, summarization, and sentiment analysis.

Following text segmentation, the process generates numerical representations termed "embeddings" for each chunk. These embeddings effectively capture the text's meaning and context, enabling various algorithms to efficiently comprehend and leverage the information. The langchain library's OpenAI Embeddings class is applied for this purpose.

C. *Storing Embeddings:*

Finally, the generated embeddings are stored along with their corresponding text chunks in a specific directory on your computer using the Chroma class. This creates a readily accessible database of text representations that can be used for various downstream applications in the future.

The below image shows a glimpse on how the embeddings are stored and the answer is retrieved using the user query.



D. *Database:*

we leverage the LangChain library, employing the vector stores module's Chroma component. Utilizing OpenAIEmbeddings to generate embeddings for a collection of documents. The resulting embeddings are then incorporated into a Chroma vector store, facilitating the creation of a comprehensive vector representation for the input documents. This approach combines the power of LangChain's vector stores and OpenAI's embeddings, offering an effective means of capturing semantic information and relationships within the document set, thereby enhancing the overall understanding and analysis of textual data.

E. *Retrieval:*

The vectordb.similarity_search() method is commonly employed to locate documents similar to a specified query within a vector database crafted using the langchain library. The breakdown is as follows:

Vector Database: This repository holds documents as numerical representations known as "embeddings," replacing the original text. These embeddings encapsulate the meaning and relationships between words in the document.

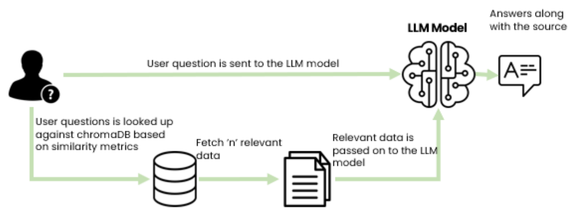
Query: You provide a document or text snippet as a query. The OpenAI Embeddings class (mentioned previously) might be used to generate an embedding for this query as well.

Similarity Search: The vectordb.similarity_search() method searches the database for documents whose embeddings are most similar to the query embedding. It calculates a distance metric (like cosine similarity) between the query embedding and each document embedding in the database.

Ranked Results: The search method returns a ranked list of documents. Documents with embeddings closest to the query embedding (smallest distance) are ranked higher, indicating higher similarity to the query content.

Finally when the answer is retrieved it is passed to the LLM model and then its frames it as into a student friendly manner making it accurate and efficient for user.

The image demonstrates the whole process in a short format. From user input to the response generated by the model.



IV. RESULTS

For assessing the effectiveness of system-generated summaries, we employed the ROUGE metric, an acronym for Recall-Oriented Understudy for Gisting Evaluation. In this context, "Gisting" refers to the extraction of the primary point from the text [31]. ROUGE serves as an evaluation matrix that compares the generated summary with a reference summary, calculating the overlap between the two concerning n-gram, word sequences, and word pairs.

A higher ROUGE score signifies a superior alignment between the summary and the reference summary. ROUGE comprises five variants, including ROUGE-N, ROUGE-S, ROUGE-L, ROUGE-W, and ROUGE-SU [32]. In this study, we specifically employed ROUGE-N and ROUGE-L metrics to analyze various summarization models, where N denotes the length of the n-gram, encompassing ROUGE-1 (unigram), ROUGE-2 (bigram), ROUGE-3 (trigram), and so forth. The definition of ROUGE-N or ROUGE-L Metrics in terms of precision, recall, and F1 Scores parameters is elucidated as follows.

Precision denotes the fraction of the summary content that is pertinent to the original document. It is computed by dividing the count of relevant sentences in the summary by the total number of sentences in the summary, as illustrated in Equation (1).

$$Precision_{ROUGE-L} = \frac{Common\ n-grams\ in\ generated\ summary\ and\ reference\ summary}{Number\ of\ \ n-grams\ in\ generated\ summary} \quad (1)$$

Recall serves as a metric for assessing the efficacy of a summarization algorithm, indicating the proportion of crucial information in the original text that is captured in the algorithm-generated summary. The equation for recall is presented in Equation (2).

$$Recall_{ROUGE-L} = \frac{Common\ n-grams\ in\ generated\ summary\ and\ reference\ summary}{Number\ of\ \ n-grams\ in\ reference\ summary} \quad (2)$$

The F1 score is a representation of the harmonic mean of precision and recall, unifying both metrics into a single score. This achieves a balanced assessment between precision and recall, as illustrated in Equation (3).

$$F1\ score_{ROUGE-L} = 2 * \frac{Precision\ *\ Recall}{Precision\ +\ Recall} \quad (3)$$

In this context, 'n' signifies the length of the n-gram under consideration (e.g., n = 1, 2, 3). These formulas are used to evaluate the quality of a summary produced by an algorithm by comparing it with a reference summary.

Table 1 presents the average precision scores of various answering techniques applied on 20 different questions. Our observation indicate that the proposed method excels, achieving a precision score of 0.91surpassing other methods.

TABLE I
AVERAGE PRECISION SCORE OF DIFFERENT MODELS

Model	**Precision**
BART	0.82
UNILM	0.84
Dense Net with Attention	0.78
Transformer with beam search	0.89
Proton Ai	**0.91**

The precision scores of 5 models are illustrated in the below figure.

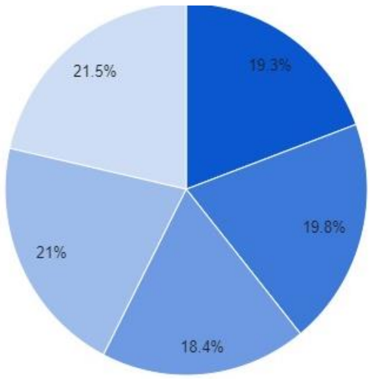


Table II displays the Average Recall score for various questions posed to the proposed model. When we examine the table it is evident that the proposed model has the highest recall stating that the words from the generated responses match nearly with the reference answer.

TABLE II
AVERAGE RECALL SCORE OF DIFFERENT MODELS

Model	Recall
BART	0.79
UNILIM	0.80
Dense Net With Attention	0.84
Transformer With beam search	0.87
ProtonAi	0.88

The recall scores of 5 different models are illustrated in the below image of line graph.

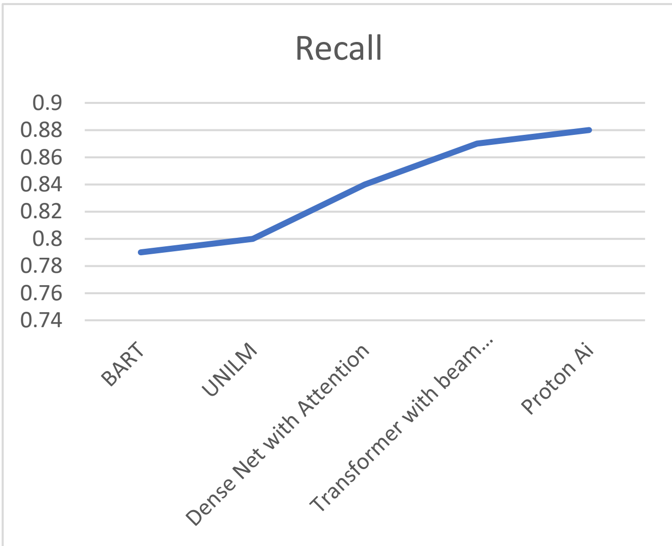
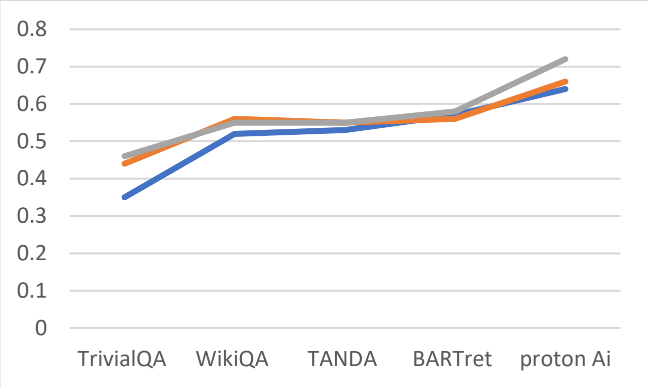


Table 3 displays the Average F1 scores of the summarizer models concerning ROUGE-1, ROUGE-2, and ROUGE-N for a given set of question. According to the table, the protonAi achieves the highest ROUGE scores.

TABLE III
AVERAGE ROUGE SCORES OF DIFFERENT MODELS

Method	**ROUGE-1**	**ROUGE-2**	**ROUGE-N**
TrivialQA	0.35	0.44	0.46
WikiQA	0.51	0.56	0.55

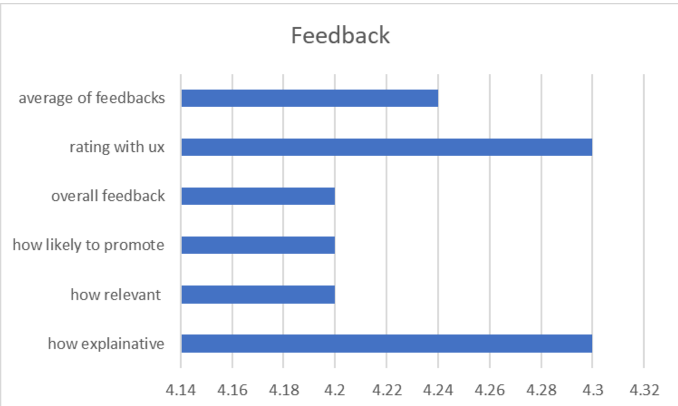
TANDA	0.53	0.55	0.55
BARTret	0.57	0.56	0.58
proposed	0.64	0.66	0.72



As a major part of our research we also conducted a human evaluation by giving access to the protonAi for students and professors and collected their feedback on terms like the relevance of the answer, explainativeness of the answer generated, and also the chances of the user for him/her to promote the method to others. Table 4 shows the details of the human evaluation.

TABLE IV
HUMAN EVALUATION

Use Case	**Value**
how explainative	4.3
how relevant	4.2
how likely to promote	4.2
overall feedback	4.2
rating with ux	4.3
average of feedbacks	**4.24**



When seen the figure above it clearly tell that users who used the model when given access seem to be greatly satisfied with the results that protonAi has generated, showing that proposed method excels in its task when compared to the previous methods.

V. CONCLUSION

The research has proven that question answering of the ProtonAi compared to the already existing models has been more accurate and efficient. Unlike the previous models having the limitations of answering the question and bit out to context and having the legitimate response guarantee at stack ProtonAi overcomes these by only restricting its knowledge base to a specific text-book knowledge. Through the implementation of the langchain retrievals and the Openai's embeddings the storing of data in the vector databases makes the response generation makes the model excel in it performance.

VI. REFERENCES

[1]Matthew Dunn, Levent Sagun, Mike Higgins, V. Ugur Guney, Volkan Cirik, Kyunghyun Cho, arXiv:1704.05179v3 [cs.CL] 11 Jun 2017

[2]Dubey Harish, Tamore Hardik , Padhi Sagar , Prof. Manisha Bharambe ,VOLUME: 07 ISSUE: 05 | MAY 2020, WWW.IRJET.NET

[3]Mandar Joshi, Eunsol Choi, Daniel S. Weld , Luke Zettlemoyer ,arXiv:1705.03551v2 [cs.CL] 13 May 2017

[4]Jonathan Berant , Andrew Chou Roy Frostig , Percy Liang ,Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 1533–1544,Seattle, Washington, USA, 18-21 October 2013. c 2013 Association for Computational Linguistics

[5]Andrea Madotto, Pascale fung, arXiv:2202.03629v5 [cs.CL] 7 Nov 2022 , Survey of Hallucination in Natural Language Generation

[6]Jared Kaplan , Benjamin Mann ,arXiv:2005.14165v4 [cs.CL] 22 Jul 2020,

[7]Rohan Kumar, Youngmin Kim , Sunitha Ravi ,KnowledgeNLP-KDD'23, August 07, 2023, Long Beach, CA

[8]Ms. Khushbu Khandait et al. / Indian Journal of Computer Science and Engineering (IJCSE)

[9]Nithya M , Madavaraj B , Sanjeev Pranesh B , Kruthikaran V ,International Journal of Innovative Science and Research Technology , ISSN No:-2456-2165 , Volume 7, Issue 10, October – 2022

[10]Puneeth Thotad , Shanta Kallur , Sukanya Amminabhavi (),Journal of Pharmaceutical Negative Results ¦ Volume 13 ¦ Special Issue 10 ¦ 2022

[11]Hala Abdel-Galil,Mai Mokhtar,Salma Doma(2021), IJICIS, Vol.21, No.2, 110-123,Automatic question generation based on beep learning approach

[12]Shivani G. Aithal , Abishek B. Rao1 ·,Sanjay Singh (2021) ,Automatic question-answer pairs generation and question similarity mechanism , https://doi.org/10.1007/s10489-021-02348-9

[13]Rohan Bhirangi, Smita Bhoir(2016) , Bhirangi et al., International Journal of Emerging Research in Management &Technology, ISSN: 2278-9359 (Volume-5, Issue-4)

[13]Bidyut Das , Mukta Majumder , Santanu Phadikar and Arif Ahmed Sekh4 (2021),Das et al. Research and Practice in Technology Enhanced Learning

[14]Tejas Chakankar, Tejas Shinkar, Shreyash Waghdhare, Srushti Waichal, Mrs. M. M. Phadtare(2023) , https://doi.org/10.22214/ijraset.2023.49390

[15]Aanchal Jawere, Anchal Soni, Nitesh (2018), IJCRT_193832

[16]Bidyut Das , Mukta Majumder, Santanu Phadikar3 and Arif Ahmed Sekh4 (2021),Das et al. Research and Practice in Technology Enhanced Learning

[17]Son The Nguyena, Theja Tulabandhulaa , Mary Beth Watson-Manheima (2024) , https://ssrn.com/abstract=4662955,

[18]Safnah Ali ,Daniella DiPaola ,Irene Lee, Jenna Hong, CHI '21, May 8–13, 2021, Yokohama, Japan

[19]Dr. Manoj Kumar, Abhishek Singh , Arnav Kumar, Ankit Kumar , 021 Fourth International Conference on Computational Intelligence and Communication Technologies (CCICT) | 978-1-6654-2392-2/21/$31.00 ©2021 IEEE | DOI:10.1109/CCICT53244.2021.00014

[20]Philipp Hacker ,Andreas Engel , Marco Mauer(2023), FAccT '23, June 12–15, 2023, Chicago, IL, USA , Fairness, Accountability, and Transparency (FAccT '23), June 12–15, 2023,Chicago, IL, USA. ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3593013.3594067

[21]Vahid Ashrafimoghari ,Necdet Gürkan , Ashrafimoghari, Vahid and Gürkan, Necdet, Evaluating Large Language Models on the GMAT: Implications for the Future of Business Education (January 1, 2024). Available at SSRN: https://ssrn.com/abstract=4681307 or http://dx.doi.org/10.2139/ssrn.4681307

[22]Deep Ganguli, Danny Hernandez, Liane Lovitt, Predictability and Surprise in Large Generative Models, FAccT '22, June 21–24, 2022, Seoul, Republic of Korea

[23]]Stefan Feuerriegel ,Jochen Hartmann ,Christian Janiesch ,Patrick Zschech(2023) , S. Feuerriegel et al.: Generative AI, Bus Inf Syst Eng

[24]Markus Freitag ,Al-Onaizan, , http://arxiv.org/abs/1910.13461 , Beam Search Strategies for Neural Machine Translation

[25]Li Dong, Nan Yang , Wenhui Wang , Furu Wei , Xiaodong Liu Yu , Wang Jianfeng Gao , Ming Zhou , Hsiao-Wuen Hon,arXiv:1905.03197v3 [cs.CL] 15 Oct 2019

[26]Mike Lewis, Yinhan Liu*, Naman Goyal, Marjan Ghazvininejad,Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, Luke Zettlemoyer, arXiv:1910.13461v1 [cs.CL] 29 Oct 2019.

[27]Gao Huang,Zhuang Liu,Laurens van der Maaten ,DOI 10.1109/CVPR.2017.243