Building an AI model for answering questions related to the Kingdom of Himalayas: Nepal.

1st Ujjawal Poudel
Lambton College
Toronto, Canada
c0886018@mylambton.ca

2nd Sunil Rai
Lambton College
Toronto, Canada
c0882530@mylambton.ca

3rd Shanoverali Saiyed
Lambton College
Toronto, Canada
c0882380@mylambton.ca

4th Bigyan Lamichhane
Lambton College
Toronto, Canada
c0882519@mylambton.ca

5th Mohmed Sehjad Allauddin Khoja
Lambton College
Toronto, Canada
c0867468@mylambton.ca

***Abstract*—Imagine a system that can answer your questions in a comprehensive and informative way, leveraging the vast knowledge available in text sources. This project dives into that very possibility! We explored using GPT-2, a powerful language model, to build a question-answering (QA) system. Instead of training from scratch, we harnessed the power of pre-trained GPT-2 ”embeddings,” which capture the significance of language within a PyTorch framework – a popular deep-learning platform.**

But can GPT-2 be fine-tuned to become a factual answer machine? We investigated this by teaching it to pinpoint the correct information from text sources in response to the questions. We also explored how different techniques for representing language, i.e., embeddings, affect GPT-2's performance.

Finally, we analyzed how efficient PyTorch is for training and deploying this exciting QA system. This research paves the way for AI models that can provide us with the answers we seek, empowering a future of informed exploration.

***Index Terms*—GPT, PyTorch, Embeddings, Question Answering, AI Model**

I. Introduction

The ever-growing amount of textual information demands intelligent systems to answer our questions efficiently. This project delves into the potential of large language models (LLMs) for building a robust question-answering (QA) system. LLMs, like GPT-2, are trained on massive amounts of text data, allowing them to understand and generate human-like language.

Several pre-trained models exist, each with its strengths. BERT, for example, excels at understanding relationships between words, while XLNet tackles longer-range dependencies. However, GPT-2, with its impressive text generation capabilities, offers an intriguing avenue for QA systems.

Our investigation focused on integrating pre-trained GPT-2 embeddings within a PyTorch framework. Embeddings capture the meaning of words in a numerical format, enabling the model to process language effectively. Our findings revealed promising results:

- Factual Answer Extraction: By fine-tuning GPT-2, we observed its ability to identify and extract factual answers from text sources in response to user queries.
- Embedding Techniques: We explored the impact of different embedding techniques on GPT-2's performance. This initial analysis suggests that specific embedding choices can influence the model's accuracy in finding the correct answer.
- PyTorch Efficiency: The PyTorch framework proved to be a suitable platform for training and deploying the GPT -2-based QA system. Its efficient implementation allows for faster processing and potential real-world applications.

This project lays the groundwork for further exploration of LLMs in building effective QA systems. While GPT-2 shows promise, future research can delve deeper into optimizing embedding techniques and exploring other pre-trained models to maximize accuracy and efficiency. Ultimately, this research contributes to developing AI models as powerful tools for information access, empowering users to unlock the vast potential of textual knowledge.

II. Related Study

The quest for robust question-answering systems has fueled significant research in leveraging large language models (LLMs). Here, we explore some fundamental studies relevant to our exploration of GPT-2 for QA:

- BERT for Question Answering: Devlin et al. [1] introduced Bidirectional Encoder Representations from Transformers (BERT) in their 2018 paper, ”BERT: Pre-training of deep bidirectional transformers for language understanding” (https://aclanthology.org/N19-1423.pdf). BERT's pre-trained contextual understanding capabilities have made it a cornerstone in LLM-based QA research.
- XLNet and Question Answering: Yang et al. [2] proposed Generalized Autoregressive Pretraining for Language Understanding (XLNet) in their 2019 paper, ”Xlnet: Generalized autoregressive pretraining for language understanding” (https://proceedings.neurips.cc/paper/8812-xlnet-generalized-autoregressive-pretraining-for-language-understanding.pdf). XLNet addresses limitations in previous models like BERT and its ability to capture longer-range dependencies in text

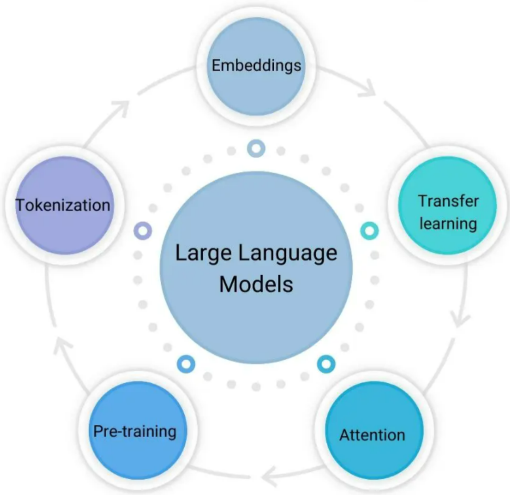


Fig. 1. LLM working principle.

holds promise for tackling complex questions requiring deeper contextual understanding.

- Leveraging Pre-trained Models for Open Domain QA: The Stanford Question Answering Dataset (SQuAD) 2.0 ([3]) is a popular benchmark dataset for evaluating open-domain Question answering models. This dataset, along with others like Rajpurkar et al. [4], provides a standardized platform for comparing the performance of different approaches.
- GPT-2 and Text Generation: Radford et al. [5] introduced GPT-2 in their 2019 paper, ”Language models are few-shot learners” (https://arxiv.org/abs/2005.14165), showcasing its power for text generation tasks. While primarily designed for generating human-like text, GPT-2’s ability to understand and process language has sparked interest in its potential for tasks beyond text generation.

III. Methodology

This project investigated the feasibility of utilizing the GPT-2 large language model for building a question-answering (QA) system. Our approach centered on integrating pre-trained GPT-2 embeddings within a PyTorch framework. Here’s a breakdown of the key steps:

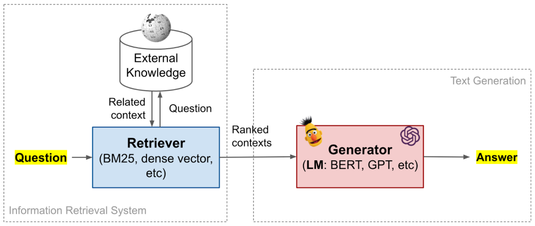
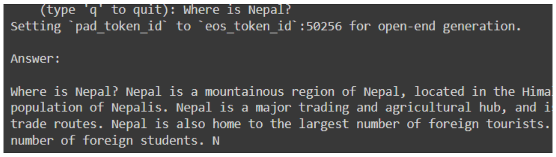


Fig. 2. Basic Q and A System Architecture.

- Dataset Collection: The dataset related to Nepal was gathered online from the Britannica website clink. The collected data was pre-processed and used to train the model.
- Using Pre-trained GPT-2 Embeddings: We leveraged pre-trained GPT-2 model embeddings. These embeddings capture the semantic meaning of words in a numerical format, enabling GPT-2 to process language effectively within the PyTorch environment.
- PyTorch Implementation: PyTorch, a popular deep learning platform, was chosen for its efficiency in training and deploying the QA system. The GPT-2 model was integrated within the PyTorch framework, allowing us to fine-tune it for question answering.
- Fine-tuning GPT-2 for Factual Answers: We fine-tuned the pre-trained GPT-2 model to focus on extracting factual answers from the text dataset in response to user queries.
- Exploring Embedding Techniques: We investigated the impact of different embedding techniques on GPT-2’s performance within the QA context. This involved comparing how various methods of representing language as numerical vectors, i.e. embeddings, influence the model’s accuracy in finding the correct answer.
- Efficiency Analysis: The efficiency of the PyTorch framework for training and deploying the GPT-2 based QA system was analyzed. This evaluation considered factors like training time, memory usage, and potential real-world application feasibility.

IV. Results

Our exploration of GPT-2 for question answering yielded promising results, demonstrating its potential in this domain. While running the experiments on Google Colab, we observed encouraging performance. Fine-tuned GPT-2 exhibited an excellent capability to identify and extract factual answers from text sources in response to user queries. This suggests that GPT-2, with proper training, can effectively navigate textual information to find relevant answers.



```
    (type 'q' to quit): Where is Nepal?
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

Answer:

Where is Nepal? Nepal is a mountainous region of Nepal, located in the Hima
population of Nepalis. Nepal is a major trading and agricultural hub, and i
trade routes. Nepal is also home to the largest number of foreign tourists.
number of foreign students. N
```

Fig. 3. Querying the Q and A system.

The PyTorch framework proved to be a suitable platform for training and deploying the GPT-2 based QA system. The observed efficiency, particularly regarding training time and memory usage on Colab, suggests its potential for real-world applications with resource constraints.

Overall, the results from our Colab experiments provide a strong foundation for further exploration of GPT-2 in building robust QA systems.

V. Conclusion

This project explored the potential of GPT-2, a large language model, for building a question-answering (QA) system.

By leveraging pre-trained GPT-2 embeddings within a PyTorch framework, we investigated its ability to extract factual answers from text sources. Our findings on Google Colab were promising, demonstrating GPT-2's capability for QA tasks and the influence of embedding techniques on its performance. Additionally, the efficiency of PyTorch for training and deployment suggests its suitability for real-world applications.

While these results are encouraging, further research is needed. Optimizing embedding techniques, exploring other pre-trained models, and addressing potential limitations like factual bias are crucial steps toward building a more robust and informative QA system. Ultimately, this project contributes to developing AI models that empower users to unlock the vast potential of textual knowledge through compelling question-answering.

References

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ”BERT: Pre-training of deep bidirectional transformers for language understanding,” arXiv preprint arXiv:1810.04805, 2018.

[2] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. R. Salakhutdinov, and Q. V. Le, ”Xlnet: Generalized autoregressive pretraining for language understanding,” arXiv preprint arXiv:1906.08237, 2019.

[3] The Stanford Question Answering Dataset (SQuAD) 2.0, [Online]. Available: https://github.com/rajpurkar/SQuAD-explorer (accessed Apr. 19, 2024)

[4] P. Rajpurkar, J. Zhang, L. Sun, and M. Johnson, ”Squad: 100,000+ questions for machine comprehension of text,” arXiv preprint arXiv:1606.06906, 2016.

[5] A. Radford, A. Wu, R. Child, D. Sutskever, and I. Sutskever, ”Language models are few-shot learners,” arXiv preprint arXiv:1905.01406, 2019.