Final Project: Development and Comparative Analysis of Generative Chatbots Using BERT and BART Architectures and Designe on the Stanford Question Answering Dataset

University of San Diago

Natural Language Processing (AAI-520-02)

Trevor Mcgirr

Eyoha .G

Reed Oken

**Abstract**

In this project, we aimed to design and implement a generative-based chatbot capable of conducting question and answer style conversations, adapting to different contexts, and handling a variety of topics. We utilized the Stanford Question Answering Dataset (SQuAD) and experimented with two state-of-the-art architectures, BERT (Bidirectional Encoder Representations from Transformers) and BART (Bidirectional and Auto-Regressive Transformers), for this purpose. Our results showed that the BERT model, when fine-tuned on the SQuAD dataset and coupled with Pinecone as the vector database, outperformed the BART model in terms of the evaluation metrics used, including BLEU and ROUGE scores. Consequently, we implemented the final chatbot using the BERT model.

**Introduction**

Chatbots have revolutionized the way we interact with technology, providing a more natural and interactive means of communication. With the integration of artificial intelligence, chatbots have evolved into intelligent conversational agents capable of understanding and responding to user queries. The goal of this project was to build a chatbot that could handle question and answer style prompts that are adapt to context (provided by the user), and handle a variety of topics, focusing primarily on our fine-tuned data set. Our SQuAD data set is a popular dataset in the natural language processing community, consisting of questions posed by crowdworkers on a set of Wikipedia articles, with the answer to every question being a segment of text from the corresponding reading passage. By leveraging this dataset, we aimed to develop a chatbot that could provide accurate and relevant answers to user queries.

**Data Cleaning/Preparation**

The SQuAD dataset was imported and preprocessed to format it correctly for training. This involved extracting question-answer pairs and their corresponding contexts, checking for any missing values, and handling any data cleaning necessary to make the dataset suitable for training the model. The data was then transformed into a format that could be used to train the BERT and BART models. The preprocessing steps were crucial in ensuring that the dataset was free from any inconsistencies or errors that could affect the performance of the models. Moreover, the preprocessing steps also helped to structure the data in a way that facilitated the training process, ultimately leading to better model performance.

**Exploratory Data Analysis**

Exploratory data analysis (EDA) was conducted to understand the characteristics of the dataset. This included analyzing the distribution of question lengths, answer lengths, and context lengths(Appendix A,B,C). Word clouds were generated for questions, answers, and contexts to visualize the most common words and identify key topics and themes (Appendix D). The results of the EDA provided valuable insights that informed the model training and evaluation process. For instance, the analysis of question and answer lengths helped to determine the appropriate input size for the models, while the word clouds helped to identify common themes and topics that could be used to evaluate the chatbot's performance.

**Model Selection**

Two state-of-the-art models were considered for the project: BERT (Bidirectional Encoder Representations from Transformers) and BART (Bidirectional and Auto-Regressive Transformers). Both models represent the cutting edge in natural language processing and have been successful in various tasks.

BERT is a transformer-based model that has revolutionized the field of natural language processing with its powerful representation learning capabilities. It has demonstrated superior performance in various natural language understanding tasks such as sentiment analysis, text summarization, and question-answering. BERT's architecture allows it to capture the context of the input text and generate responses that are highly relevant to the query. The decision to use BERT was based on its proven effectiveness in similar tasks and its ability to handle multi-turn conversations, which was a key requirement for our chatbot.

On the other hand, BART is a state-of-the-art model specifically designed for sequence-to-sequence tasks such as text generation and question-answering. BART's architecture combines the benefits of autoregressive models and denoising autoencoders, making it an ideal choice for generating coherent and contextually relevant responses. The decision to experiment with BART was based on its potential to generate high-quality responses for the chatbot, given its success in similar text generation tasks.

Both models were fine-tuned on the SQuAD dataset, with BERT being coupled with Pinecone as the vector database to enhance its performance further. Pinecone, a vector database, provides a convenient and efficient way to store and retrieve high-dimensional vectors, which is essential for capturing the semantic relationships between words and phrases. The integration of Pinecone with BERT aimed to leverage these semantic relationships to generate more accurate and relevant responses.

**Model Analysis**

The BERT model achieved impressive results when fine-tuned on the SQuAD dataset and coupled with Pinecone as the vector database. The BLEU scores ranged from 0.107 to 1.030, and the eval loss consistently decreased from -5.30 to -7.67, indicating that the model was learning effectively and generating high-quality responses. These results highlight the effectiveness of BERT in capturing the context and generating relevant responses, which is crucial for a question-answering chatbot. The integration of Pinecone further enhanced the model's performance by leveraging the semantic relationships between words and phrases, leading to more accurate and contextually relevant responses.

In comparison, the BART model faced some challenges during the training process, which ultimately affected its performance. The average BLEU score was 0.0001, and the average training and validation losses were 1.81 and 2.88, respectively. Despite various fine-tuning approaches taken to improve the BART model's performance, such as adjusting the learning rate, batch size, and number of training epochs, the results were not as satisfactory as those achieved by the BERT model. We even tried to a few advanced approached such as;

Early Stopping which monitor the validation loss and stop training once it stops decreasing (and starts increasing),

Regularization, which adds regularization terms to the loss function to penalize large weights.

Implement Dropout to add dropout layers to the model to randomly zero out some of the weights during training, which can prevent overfitting.

These challenges highlight the limitations of BART in handling the complexity of the SQuAD dataset and the requirements of the chatbot.

The decision to implement the final chatbot using the BERT model was based on its superior performance and the potential benefits of using Pinecone as the vector database. However, it is important to note that the BERT model is not without its challenges. Fine-tuning BERT requires careful consideration of various hyperparameters such as learning rate, batch size, and number of training epochs, which can significantly affect the model's performance. Moreover, BERT's architecture is complex and requires substantial computational resources, which can be a limitation for some applications. Despite these challenges, the benefits of using BERT, such as its powerful representation learning capabilities and proven effectiveness in natural language processing tasks, outweigh the risks.

The analysis of the BERT and BART models revealed distinct differences in their performance and the challenges faced during fine-tuning.

**Conclusion**

In conclusion, the BERT model, when fine-tuned on the SQuAD dataset and coupled with Pinecone as the vector database, outperformed the BART model in terms of the evaluation metrics used. Consequently, the final chatbot was implemented using the BERT model. This report detailed the challenges faced, solutions implemented, model architecture, evaluation and results. Looking ahead we could explore integrating advanced NLP techniques such as sentiment analysis, entity recognition, and language translation could further augment the chatbot's functionality. The integration of real-time learning capabilities, where the chatbot learns and evolves from user interactions, is another exciting possibility that could be explored in future iterations of this project.

Appendix

A.

A graph of a number of words

Description automatically generated

A graph with numbers and a number of words

Description automatically generatedB.

C.

A graph of a distribution of context lengths

Description automatically generated

D.

Questions word cloud

A close-up of words

Description automatically generated

Answers word cloud

A close up of words

Description automatically generated

Context word cloud

A close up of words

Description automatically generated