Midterm Report

Enhancing Question-Answering Systems with BERT

Project Group 50

Ayushi Chakrabarty, Cameron Potter, Kshitij Pathania, Prateek Yadav, Sneha Maheshwari

{achakrabarty8, cpotter8, kpathania3, p34, smaheshwari63}@gatech.edu

Introduction/Background

In the evolving landscape of Natural Language Processing (NLP), the quest for models that can understand and generate human-like responses has become paramount. Among the myriad of models that have emerged, BERT [1] (Bidirectional Encoder Representations from Transformers) stands out as a revolutionary architecture that has set new benchmarks in a range of NLP tasks. The underlying Transformer architecture, introduced by Vaswani et al. [5], forms the basis of BERT, bringing a significant shift in how deep learning models process sequences. Specifically, in the domain of Question-Answering (QA) systems, the adaptability and prowess of BERT can lead to significant enhancements in accuracy, contextual understanding, and response generation. This paper delves deep into the nuances of integrating BERT into QA tasks, showcasing its potential to reshape the way machines understand and answer questions. As we navigate the landscape of deep learning, this integration aligns with the evolutionary trajectory outlined by Goodfellow, Bengio, and Courville [2], marking a transformative step towards endowing machines with more human-like conversational abilities.

Problem Definition

The primary challenge is to investigate and evaluate how BERT can be effectively integrated into existing or novel QA frameworks to enhance their performance. This includes:

1. How can BERT's bidirectional understanding of context be harnessed to improve the accuracy of QA systems?
2. What modifications or fine-tuning techniques are required to adapt BERT specifically for diverse QA tasks, considering the variability in question types and domains?
3. How can the scalability and efficiency of QA systems be maintained or improved upon integrating the computationally intensive BERT model?
4. In what ways can BERT's capabilities be leveraged to make QA systems more robust against ambiguous, misleading, or poorly framed questions?

Data Collection

The data for this project was obtained from the Stanford Question Answering Dataset [3] (SQuAD 2.0). This dataset includes a set of questions and answers based on a series of Wikipedia articles. We used two subsets of this dataset: the training set (train-v2.0.json) and the development set (dev-v2.0.json). These files were downloaded and loaded into our environment for processing and training our model. Preprocessing steps included tokenizing contexts and questions and mapping answer start and end positions in the token space. In addition to preprocessing, we analyzed the distribution of the lengths of contexts and questions to gain insight into the dataset. The length of a context and its corresponding questions can significantly impact the complexity of the QA task.

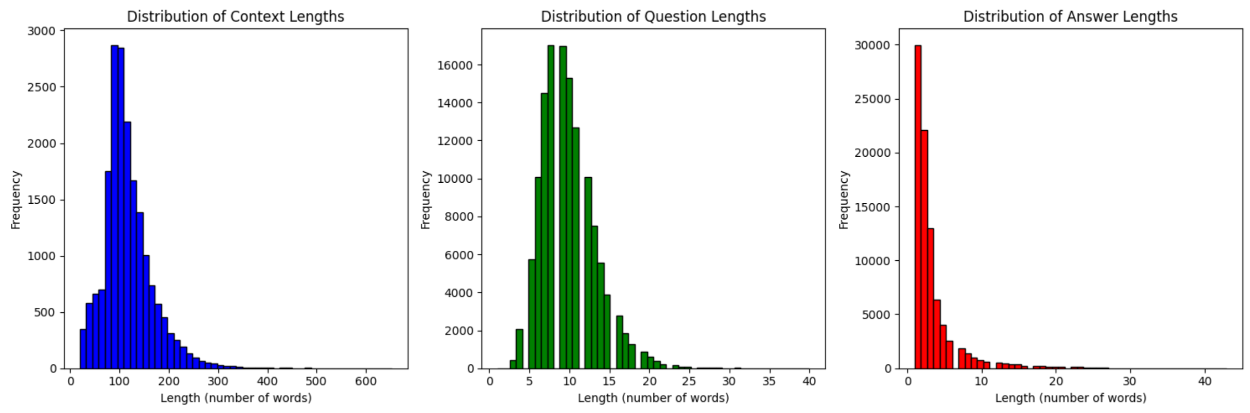


Figure 1: Distribution of context, question and answer lengths in the SQuAD 2.0 training set.

Methods

Our methodology involved a series of steps from data loading and preprocessing to model training and fine-tuning. The following details each step:

Data Loading and Preprocessing

We loaded our data from the SQuAD v2.0 dataset, which includes a training set (`train-v2.0.json`) and a development set (`dev-v2.0.json`). The dataset consists of contexts, questions, and their corresponding answers. The preprocessing steps included:

- Parsing each context, question, and answer from the dataset.
- Using the `AutoTokenizer` from the Hugging Face Transformers library to tokenize the contexts and questions. We chose the "distilbert-base-uncased" model for tokenization.
- Encoding the answers by mapping their character start and end positions to token positions.
- Handling cases where the answer passages were truncated due to tokenization limits.

Dataset Preparation

A custom dataset class was implemented to manage the tokenized encodings. This class provided mechanisms to retrieve individual data points and their lengths, crucial for creating batches during training and evaluation.

Model and Training

For the question-answering task, we used the `AutoModelForQuestionAnswering` from the Hugging Face library, again utilizing the "distilbert-base-uncased" model. We did the manual tuning for identifying optimal hyperparameters. The training process was carried out with the following settings:

- Hyperparameters: Number of Epochs (`N_EPOCHS`) = 5, Learning Rate = 5e-5, Weight Decay = 0.01, Batch Size = 16.
- The optimizer used was AdamW, adhering to the specified learning rate and weight decay.
- Training involved iterating over batches of the training dataset, performing forward and backward passes, and updating the model parameters.
- We tracked the loss at each step, which can be visualized to understand the model's learning progress over epochs.

Below is a line plot showing the training loss over epochs.

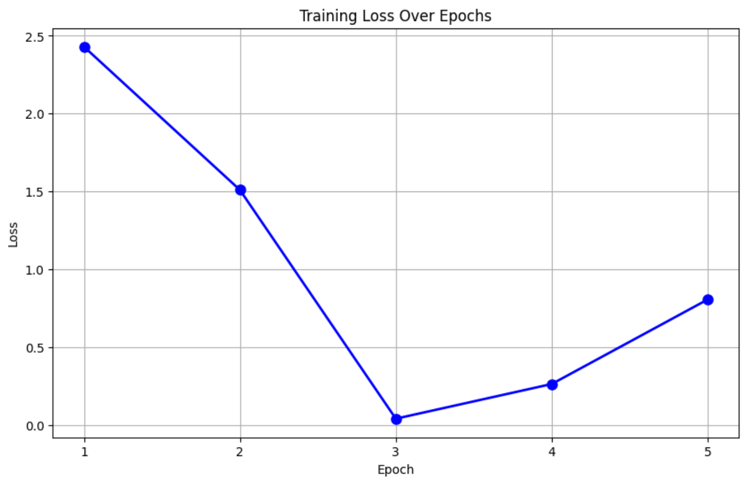


Figure 2: Training Loss Over Epochs

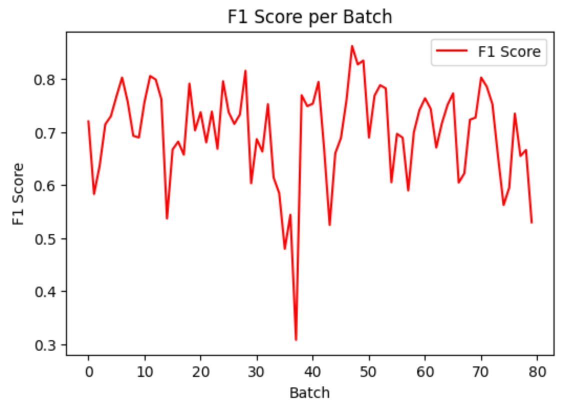
The loss significantly decreases from Epoch 1 to Epoch 3, dropping from 2.43 to 0.04. This is a strong indication that the model is learning effectively and improving in its ability to make accurate predictions. However, after Epoch 3, there is an increase in loss - from 0.04 in Epoch 3 to 0.263 in Epoch 4, and further to 0.806 in Epoch 5. The increase in loss could be indicative of overfitting, where the model starts to memorize the training data, reducing its ability to generalize to new, unseen data. Thus we stop our training in the epoch 5 to avoid overfitting and saved the weights of our model at this stage.

Results and Discussion

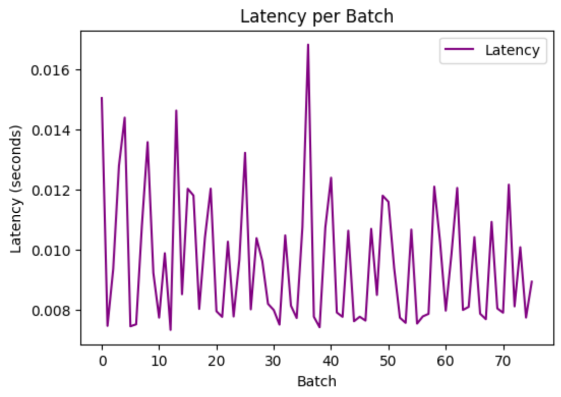
The evaluation of our fine-tuned BERT model on the test dataset provided insightful metrics into its performance. We observed substantial accuracy in predicting the start and end positions of answers, which are critical for the efficacy of Question-Answering systems. The F1 score was computed across batches, offering a comprehensive measure that captures both the precision and recall of the model's predictions.

In addition to accuracy metrics, we monitored the latency of the model's responses, which is pivotal for real-time applications. Initially, a significant outlier in the latency measurements was identified, which skewed the overall representation of the model's responsiveness. After adjusting for these outliers, we obtained a more representative overview of the model's average response time.

These metrics were visualized in a series of plots, providing a clear depiction of the model's performance dynamics across evaluation batches. Below, Figures 3a and 3b display the F1 Score and Latency over batches, respectively.



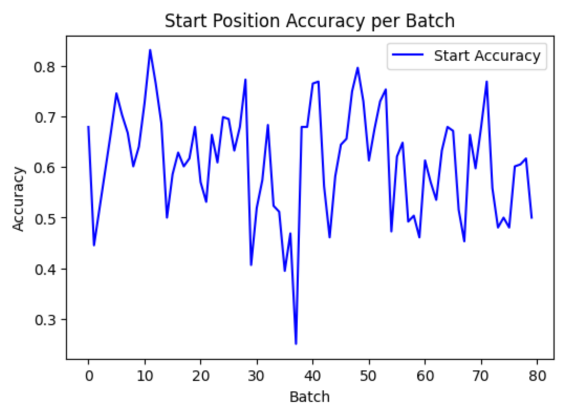
(a) F1 Score per batch



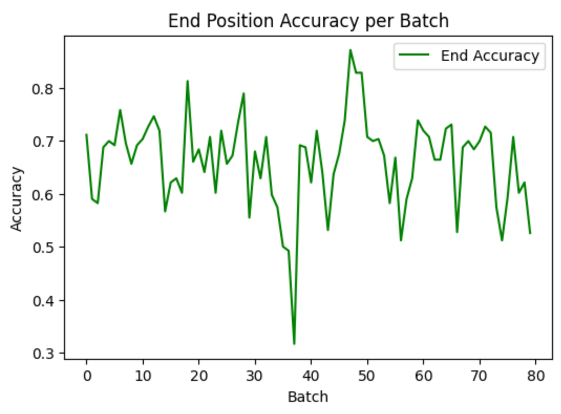
(b) Adjusted Latency of Responses per batch

Figure 3: Evaluation Metrics for Fine-tuned BERT: F1 Score and Latency

Figures 4a and 4b show the Start and End Position Accuracy per batch, illustrating the model's precision in identifying the correct span of text that answers the posed questions.



(a) Start Position Accuracy per batch



(b) End Position Accuracy per batch

Figure 4: Start and End Position Accuracy of the Fine-tuned BERT Model

Adding to the above, we have summarized our cumulative results as follows:

- **Start Accuracy:** Our model achieved a start accuracy of 0.6116, indicating its effectiveness in identifying the beginning of the answer span.
- **End Accuracy:** The end accuracy was 0.6618, showing the model's ability to correctly pinpoint the end of the answer span.
- **F1 Score:** We achieved an F1 score of 0.7002, reflecting a strong balance between precision and recall in our predictions.
- **Average Latency:** After adjusting for outliers, the average latency of the model's responses was approximately 0.0086 seconds, demonstrating its suitability for real-time applications.

Pending Tasks

In the final stages of our project, we aim to focus on several key areas. Firstly, we will further investigate results and Fine-tune the BERT to better utilize BERT's bidirectional understanding of context, enhancing the accuracy and depth of responses in QA systems. Another critical area is Robustness Against Varied Queries, where we will test the system's ability to handle ambiguous, misleading, or poorly framed questions, and develop mechanisms to improve its robustness in these scenarios. In terms of Interpretability and Decision-Making Analysis, our efforts will be directed towards delving into the interpretability of BERT's decision-making process to demystify how the model arrives at its answers and enhancing the transparency of the QA system. Additionally, we will conduct a Comprehensive Performance Evaluation, involving extensive testing focusing on the robustness and adaptability of the model to a variety of question types and contexts, and utilizing a broader range of metrics for assessment, including user satisfaction and system usability. Lastly, we will explore an Expansion to Other Datasets, such as QA datasets like CommonsenseQA [4], to broaden the scope and applicability of our research.

Contribution table

Student Name	**Contributed Aspects**
Ayushi Chakrabarty	Problem Definition, Documentation, Report Writing
Cameron George Potter	Ideation of the project theme, Methods, Model Training
Kshitij Pathania	Model evaluation, Results Compilation, Data visualisation
Prateek	Problem Definition, Ideation of the project theme, Documentation
Sneha Maheshwari	Problem Definition, Report Writing, Documentation

Table 1: Contributions of team members

References

[1] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding.

[2] Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. MIT Press, 2016.

[3] Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. Squad: 100,000+ questions for machine comprehension of text, 2016.

[4] Talmor, A., and Berant, J. Commonsenseqa: A question answering challenge targeting commonsense knowledge, 2019.

[5] Vaswani, A., Shazeer, N., Parmar, N., et al. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (2017), Curran Associates Inc.