Team Number – 61

Abstract

This paper addresses the problem of developing a question answering model that provides accurate predictions on a custom dataset while also reducing prediction time. To tackle this problem, several models such as DistilBERT[1]*, ALBERT*[2]*, Intra Ensemble*[3]*, and FP Net*[4]*, were considered and evaluated. After a thorough study, it was concluded that DistilBERT was the best fit for the requirements. The model was implemented using a BERT transformer*[5] *from Hugging Face, fine-tuned on a proprietary dataset, and optimized through experiments to determine appropriate hyperparameters for optimal accuracy and training performance. Synthetic data generation was employed to further improve accuracy, and model parameters were adjusted to reduce prediction time. The Hugging Face pipeline was utilized to integrate the tokenizer with the model. Although other techniques such as model weight pruning and custom input pipeline creation were considered, they were not compatible with the current model and did not meet the task requirements. The final model showed significant improvements in both prediction accuracy and prediction time.*

Implementation

1. Training

1.1. Synthetic Data Generation

Due to restrictions in using external databases, we resorted to generating our own data by splitting each datapoint into two parts. This process of splitting the data instantly doubled the size of our dataset and made training more effective. The following paragraph is split into two separate ones, consider the example below.

Eg:

“ Heresy is any provocative belief or theory that is strongly at variance with established beliefs or customs. A heretic is a proponent of such claims or beliefs. Heresy is distinct from both apostasy, which is the explicit renunciation of one's religion, principles or cause, and blasphemy, which is an impious utterance or action concerning God or sacred things. The term is usually used to refer to violations of important religious teachings, but is used also of views strongly opposed to any generally accepted ideas. It is used in particular in reference to Christianity, Judaism, Islam and Marxism. In certain historical Christian, Islamic and Jewish cultures, among others, espousing ideas deemed heretical has been and in some cases still is subjected not merely to punishments such as

excommunication, but even to the death penalty. The term heresy is from Greek αἵρεσις originally meant "choice" or "thing chosen", but it came to mean the "party or school of a man's choice" and also referred to that process whereby a young person would examine various philosophies to determine how to live. The word "heresy" is usually used within a Christian, Jewish, or Islamic context, and implies slightly different meanings in each."

The Above Paragraph was splitted into following two paragraphs–

1. *" Heresy is any provocative belief or theory that is strongly at variance with established beliefs or customs. A heretic is a proponent of such claims or beliefs. Heresy is distinct from both apostasy, which is the explicit renunciation of one's religion, principles or cause, and blasphemy, which is an impious utterance or action concerning God or sacred things. The term is usually used to refer to violations of important religious teachings, but is used also of views strongly opposed to any generally accepted ideas. It is used in particular in reference to Christianity, Judaism, Islam and Marxism."*

2. *"In certain historical Christian, Islamic and Jewish cultures, among others, espousing ideas deemed heretical has been and in some cases still is subjected not merely to punishments such as excommunication, but even to the death penalty. The term heresy is from Greek αἵρεσις originally meant "choice" or "thing chosen", but it came to mean the "party or school of a man's choice" and also referred to that process whereby a young person would examine various philosophies to determine how to live. The word "heresy" is usually used within a Christian, Jewish, or Islamic context, and implies slightly different meanings in each."*

1.2. Training Methodology

Initially, we started with a pretrained model which was neither trained, nor fine tuned on the provided data set. The predictions were not up to the mark. Some of the parameters that we used were : `n_heads = 12, n_layers = 6, sinusoidal_pos_embds = false`. The F1 score that we got is `85.1`. Then we tried to build a custom model from scratch which was inspired by pre-existing bert architecture and was trained on the synthetic dataset. `n_heads = 8, n_layers = 6, sinusoidal_pos_embds = true`. More information in Appendix.

The results were still not satisfactory. The comparison plots are shown in the Fig. 1, 2

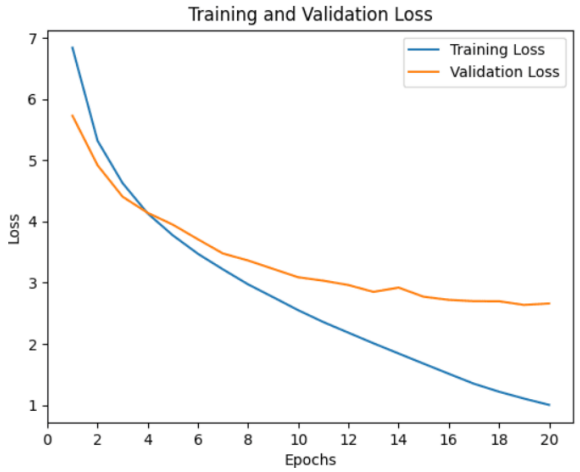

Figure 1. Training Loss vs. Epochs for Pre-trained Model, Model has not yet seen our data.

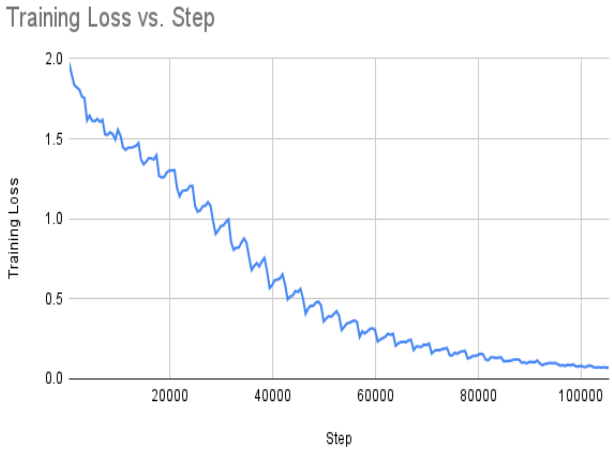

Figure 2. Training Loss vs. Steps for Model trained on synthetic dataset

1.3. Fine Tuning Methodology

Learning Rate	Weight Decay	Epochs	Training loss
2e-5	0.1	0.5	0.78
2e-5	0.01	1.0	0.23
2e-7	0.001	0.5	0.04

Table 1: Hyperparameters for Fine Tuning 2. Number of heads = 12, Input Size = 196

We utilized a pre-trained bert-based cased model for fine-tuning as it showed potential for improvement. Our input pipeline was integrated with the preprocessed dataset and used for fine-tuning the model. The tokenizer used was *"distilbert-base-cased-distilled-squad* [6]*"*.

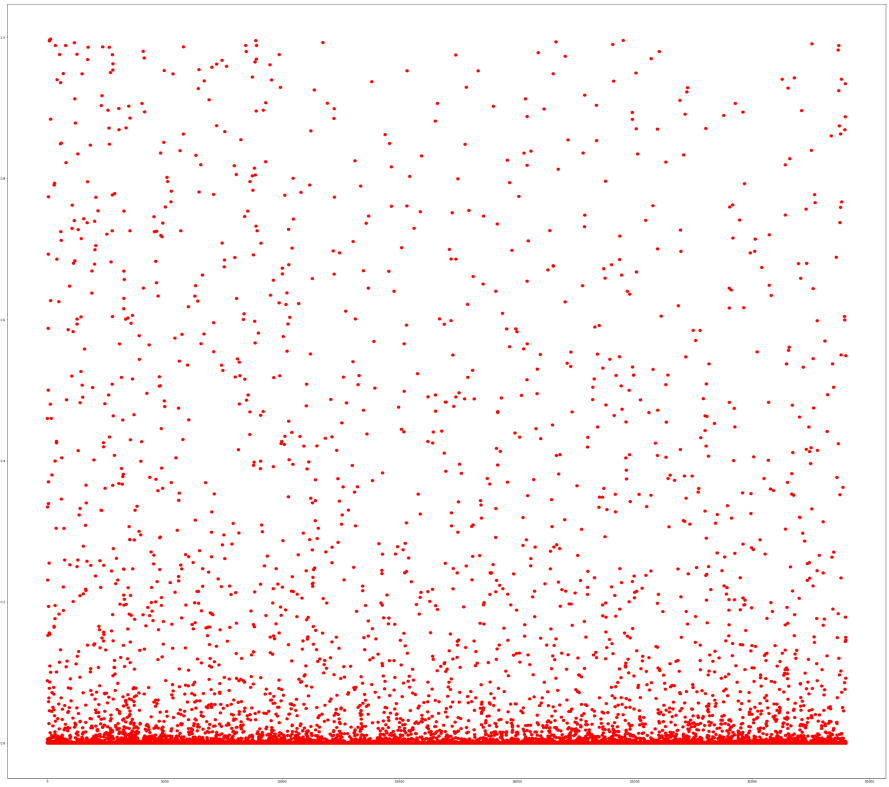

Figure 3. Plot of scores (0 - 100) vs input samples

Learning Rate	Weight Decay	Epochs	Training loss
2e-5	0.1	0.5	0.86
2e-5	0.01	1.0	0.17
2e-7	0.001	0.5	0.02

Table 2: Hyperparameters for Fine Tuning 2. Number of heads = 8, Input Size = 204

To ensure that the model did not overfit during fine-tuning, careful monitoring of the training loss versus steps was carried out. If the slope of the training loss was constant, it indicated that the model was overfitting, which was not acceptable. This monitoring helped to ensure that the model was generalizing well to the data and did not just memorize the training data.

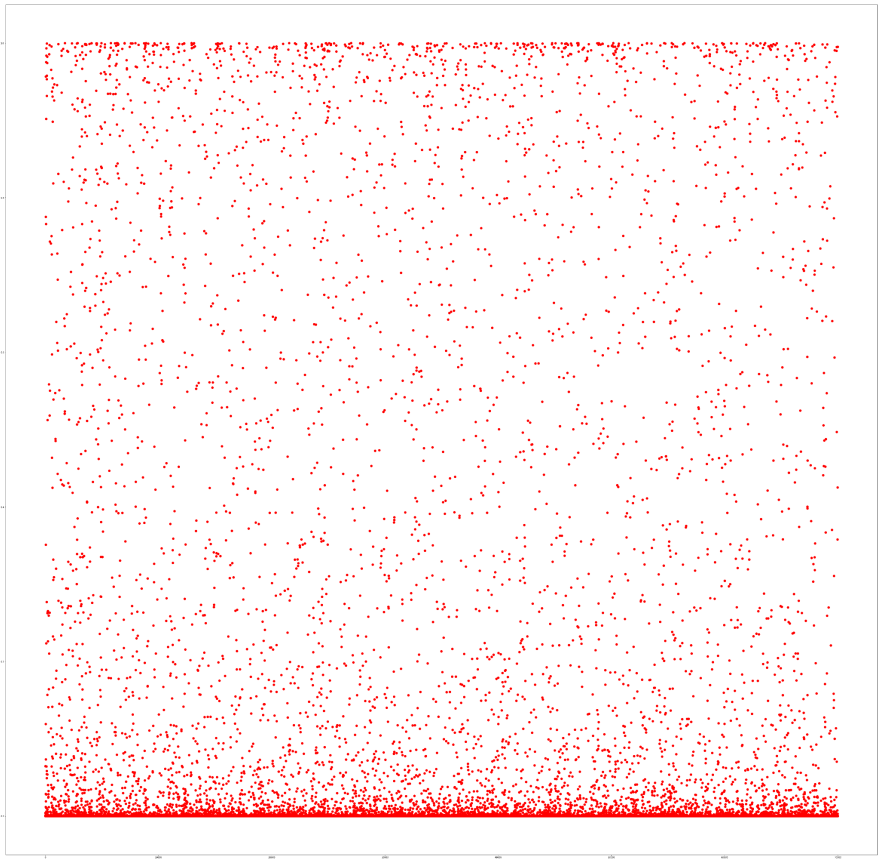

Figure 4. Plot of scores (0 - 100) vs input samples

Red dots represent False Negatives. Vertical axis denotes scores (0 - 100%) and horizontal axis denotes input samples. The density of red dots is greater at bottom denoting that the model predicts for wrong answers with low confidence, thus increased accuracy. Which is the desired outcome.

In the comparison of two fine-tuned models, it was noted that both models had good prediction accuracy, however, the second fine-tuned model had a higher density of red dots at the top. This suggests that the model makes incorrect predictions with high confidence for certain questions

with incorrect answers. Additionally, it was observed that the second model predicts faster as compared to the first model, as indicated by Fig. 5, 6, which shows improved run-time performance. This was achieved through reducing the number of attention heads and augmenting the data. The trade-off between accuracy and run-time was carefully considered and optimized to meet the project requirements.

1.4. Run Time Improvements

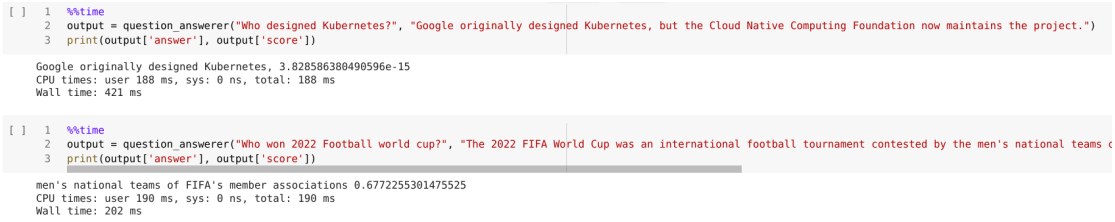

Figure 5. Runtime for Fine Tuning – 1 is **~ 400 ms**. Number of heads = 12, Input Size = 196

Figure 6. Runtime for Fine Tuning – 2 is **~ 70 ms**. Number of heads = 8, Input Size = 204.

On comparing the two Fine tuned models, we can see significant improvements in prediction times by reducing the number of attention heads. Refer to Appendix for time computation metric.

1.5. Final Implemented Pipeline

The data retrieval pipeline in our implementation consisted of constructing a dictionary that maps themes as keys to a list of paragraph-ID pairs as values. When a question is received, the associated paragraph theme is searched in the entire dictionary. Based on the theme, all pairs of paragraphs and questions related to that theme are retrieved from the database and sent for prediction. Following is the schematic description of the pipeline.

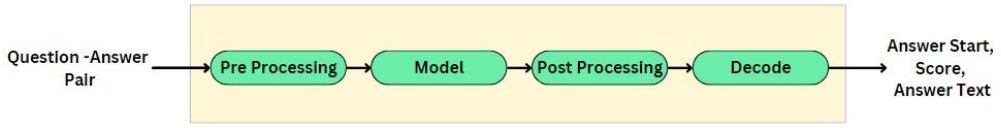

Figure 7. The final implemented pipeline for training

2. Testing

2.1. Synthetic Data Generation

The input data was initially provided in CSV format, which was not compatible with the input pipeline. To resolve this issue, the data was converted from CSV to JSON format using Node.js. This allowed for seamless integration with the input pipeline for fine tuning the model.

The custom JSON object was structured in a manner that optimized data retrieval for predictions. It had a key-value format, where the keys were questions, and the values were lists of pairs of paragraphs and paragraph IDs. Below is the sample format of JSON.

```
{
      "<question>" : [{<paragraph>, <paragraph_id>}...]
}
```

Conclusion

In this paper, a question-answering model was developed with the goal of achieving high accuracy and reducing prediction time. After evaluating several models including DistilBERT, ALBERT, Intra Ensemble, and FP Net, DistilBERT was selected for implementation. The model was fine-tuned on a proprietary dataset using Hugging Face's BERT transformer, and hyperparameters were optimized for best performance. Synthetic data generation was used to further improve accuracy. The Hugging Face pipeline was used to integrate the tokenizer with the model. The final model showed significant improvements in

both accuracy and prediction time. The fine-tuning methodology involved two steps, both using a pre-trained BERT-based cased model. In the first step, the number of heads was set to 12, and the input size was 196. In the second step, the number of heads was set to 8, and the input size was 204. The model was monitored for overfitting by comparing the training loss with the steps. The results showed that the fine-tuned model outperformed the original pre-trained model in terms of accuracy and prediction time

References

1. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter –Victor Sanh, Lysandre Debut, Julien Chaumond, Thomas Wolf – arXiv:1910.01108v4 [cs.CL] https://doi.org/10.48550/arXiv.1910.01108
2. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations – Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, Radu Soricut – arXiv:1909.11942v6 [cs.CL] https://doi.org/10.48550/arXiv.1909.11942
3. Intra-Ensemble in Neural Networks – Yuan Gao, Zixiang Cai, Lei Yu – arXiv:1904.04466v2 [cs.CV] https://doi.org/10.48550/arXiv.1904.04466
4. Grüning, P., Martinetz, T., & Barth, E. (2022). FP-nets as novel deep networks inspired by vision. In Journal of Vision (Vol. 22, Issue 1, p. 8). Association for Research in Vision and Ophthalmology (ARVO). https://doi.org/10.1167/jov.22.1.8
5. Attention Is All You Need: Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is All you Need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), Advances in Neural Information Processing Systems (Vol. 30). https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
6. https://huggingface.co/distilbert-base-cased-distilled-squad

Appendix

1. Wall time – The report considers wall time as the metric to compute time for prediction.
2. `n_heads` – This denotes the number of attention heads in the BERT model. Default value = 12, value we used finally in our model = 8
3. `n_layers` – This denotes the number of layers in the BERT. Default value = 6
4. `sinusoidal_pos_embds` – Sinusoidal positional embeddings generate embeddings using `sine` and `cosine` functions.