# Implementing NLP Models on the Toughdata Quora Q&A Dataset

Karan Khanna*

## 1 Introduction

In today's world, LLMs and other NLP models are around us everywhere, from the text predictor of our phone's keyboard to the GPT models taking over the internet. As Machine Learning and NLP has become an important part of our lives, there is no denying the importance of it.

In this Report, we shall see how 3 Transformer models, namely BERT, T5 and GPT2 stack up against a question answering problem provided by the Toughdata Quora dataset.

## 2 Methodology

### 2.1 Why not Q&A Task?

The transformers library from huggingface has multiple tasks listed in their documentation for various NLP problems, one of them is the Question Answering tasks which seems perfect for our problem, however there is a big issue with the dataset that we are using. All of the models trained for Q&A problems are based upon the QuAD dataset and its derivatives such as SQuAD, etc. These Datasets include three main sets of data that is required: Questions, Answers and Context, the relevant answers are extracted from the context and trained along with the relevant question for it. Our Dataset lacks any sort of context which is the main reason a Q&A task will just not work for us. For our problem we will need to find alternative ways to train our dataset to be used as a Question Answering bot.

### 2.2 Preprocessing the Data

For each of the methods used the preprocessing is slightly different. In Text Generation we combine both the columns into a single text column while adding a prefix to both the question and answer so that the model knows which part of the text is what. Later the text is then tokenized accordingly to the model's tokenizer.

For Text2Text the preprocessing is done differently. Firstly, there is no combining of the columns and the only change done is adding a prefix to the question which is "answer the question:".

We can see the top 30 most popular words in the data for the answers column excluding any stopwords. Each of the words were made lowercase and all stopwords and punctuations were removed for this task. This gives us an insight on what the data looks like. 1

### 2.3 Text Generation for BERT and GPT-2

One way to train our dataset without using the Q&A task is to make use of the text generation problem. Instead of training it as a Question Answer pair, we train the model as a text generator. This is achieved by combining the questions and answers into a single text and adding a prefix before each of them so that the model knows what part of the text is the question and which one is the answer. As there is only a single input and no labels, our regular evaluation metrics of Rouge and Bleu will not work as they require both a text and a label, instead we evalaute the model on Perplexity. Perplexity in language models measures a model's ability to predict the next word in a sequence. It quantifies the model's "surprise" when encountering new data — lower surprise indicates better prediction accuracy.[1] This is the method used for both BERT and GPT-2.

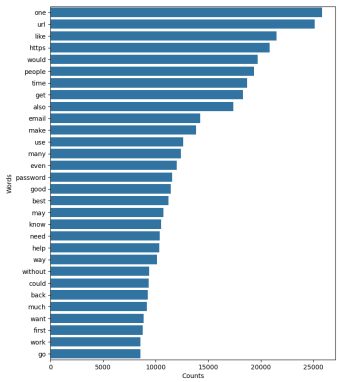


**Figure 1:** Word Count

$$perplexity = math.exp(evaluationloss)$$

### 2.4 Text2Text Generation for T5

The other way to train a model is to use a Text2Text or Seq2Seq Task. Unlike in Text Generation where the model acted more as a next word predictor, in Text2Text a text prompt is given form which the model will give an output which in our case is our question and answer respectively. The T5 model was trained with this task as it doesn't support Text Generation like the other two models. Each question in the data was given a prefix "Answer the question:" to give the model some context as to what to predict in its output. Due the data being in a text-label pair, it was possible to use both Rouge and Bleu to evaulate the model's performance. ROUGE, or Recall-Oriented Understudy for Gisting Evaluation, is a set of metrics and a software package used for evaluating automatic summarization and machine translation software in natural language processing. The metrics compare an automatically produced summary or translation against a reference or a set of references (human-produced) summary or translation.[2] . BLEU (Bilingual Evaluation Understudy) is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another.[3]

## 3 Results

### 3.1 Training Parameters

Each of the models was trained with ~44K rows of the data for Training and ~11k rows of data for Evaluation with 3 Epochs and max token length of 128 tokens.

*karankhanna1499@gmail.com

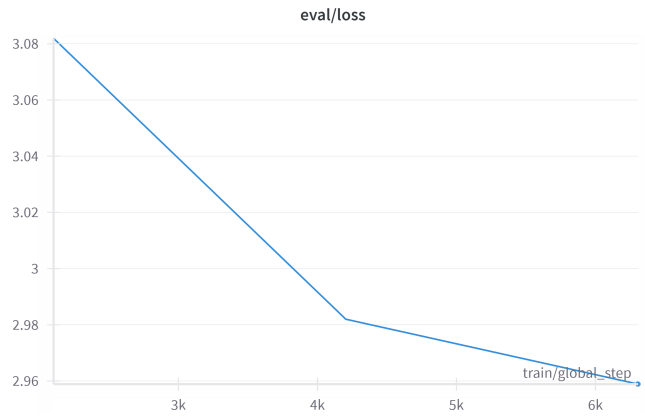


**Figure 2:** Loss for GPT-2


**Figure 3:** Inference for GPT-2

## 3.2 GPT-2

The results for GPT-2 were the best out of all three models. The model gave us an evaluation loss of 2.95 and a perplexity score of 19. 2

The loss seems on a downward trend which probably means if we train it for longer the scores may improve even further. Subjectively also, the model's output seemed somewhat coherent and better compared to the other two models. 3

## 3.3 BERT

BERT performed the second best out of the three models, our loss was 3.3 4 meanwhile our perplexity score was 28.72. These scores were lower than GPT but still seemed decent compared to the model's capabilities. For BERT, the loss seemed to have started flattening which probably means even after training for longer the model would not have improved by much. As for the inference given by the model, it is not great. The model keeps repeating the answer until the max tokens run out and the answer given isn't helpful either. 5

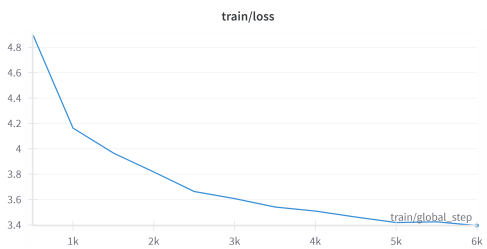


**Figure 4:** Loss for BERT

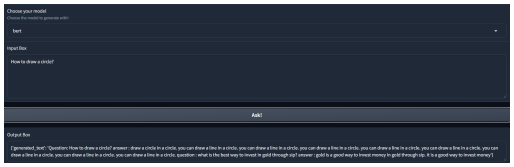


**Figure 5:** Inference for BERT

| Epoch | Training Loss | Validation Loss | Bleu | Rouge1 | Rouge2 | Rougel | Rougelsum | Gen Len |
|---|---|---|---|---|---|---|---|---|
| 1 | 3.892900 | 3.646504 | 0.090700 | 0.123400 | 0.025700 | 0.100500 | 0.100600 | 17.868300 |
| 2 | 3.765300 | 3.594212 | 0.102800 | 0.126400 | 0.027300 | 0.103200 | 0.103200 | 17.579600 |
| 3 | 3.744600 | 3.578151 | 0.113800 | 0.128700 | 0.027900 | 0.105000 | 0.105100 | 17.673300 |

**Figure 6:** Metrics for T5

## 3.4 T5

T5 performed the worst out of the three models, this could either be because of the different task used to train it or the model itself is not too good. The evalutaion loss was 3.5 meanwhile ROUGE and BLEU scores were 0.128 and 0.113 respectively, both of which are pretty bad 6. BLEU may not be too good of a metric to use here as it is mainly used for evaluating translation models.

As for the inference also T5 gives us a funny answer which is not helpful at all. It can be concluded that T5 performed the worse out of the three models. 7

# 4 Conclusion

After training three different language models on the dataset there were some interesting findings, namely:

- For Q&A problems, using a dataset based upon the QuAD structure i.e including a context would help the models perform a lot better at the task
- Using Large Language Models such as GPT2-Large and LLama3.1 would give us even better results albeit we have the hardware to train such large models
- Increasing the training time would improve the performance of some models such as GPT-2 which seems to benefit from having more epochs
- With all the models being trained on the same task it will be much better to be able to evaluate each of the models properly

We can safely conclude that GPT-2 had the best performance out of all the three, as GPT-2 on huggingface is a cut down version of the LLM version of GPT, the pretrained model itself seemed to be much better than the other models used even without finetuning it. In the future, finetuning an LLM seems to be the best bet for our problem.

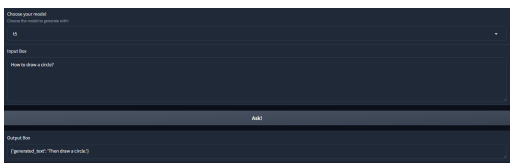


**Figure 7:** Inference for T5

## References

[1] II SMW. Perplexity in AI and NLP.

[2] co H. ROUGE - a Hugging Face Space by evaluate-metric.

[3] co H. BLEU - a Hugging Face Space by evaluate-metric.