THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

Divya Parmar Individual Report
DATS 6312
December 9th, 2021

Introduction

For our final project, I worked with Tristin Johnson and Robert Hilly on the GLUE tasks. GLUE stands for General Language Understanding Evaluation, and these tasks cover various concepts in natural language processing.

There are 11 tasks, and we split them among us. In addition, we worked on the PowerPoint presentation and final report.

Description of My Work

I was assigned MNLI Matched, MNLI Mismatched, QNLI, and Diagnostics.

- **MNLI Matched:** Each observation contains a premise ("Your gift is appreciated by each and every student who will benefit from your generosity.") and a hypothesis ("Hundreds of students will benefit from your generosity."). Here, we're interested in predicting if the hypothesis logically flows from the premise. Each pair is either classified as entailment (it does flow), neutral, or contradiction (premise shows the opposite of hypothesis).
- **MNLI Mismatched:** Trained on the same MNLI dataset as MNLI Matched, but has a different validation set of mismatched sentences.

- **QNLI:** This task contains a question and a sentence taken from a context paragraph, with the intent to see if the sentence answers the question.
- **Diagnostics:** This task is the same as MNLI. However, I did not do this task as the labels were provided by the HuggingFace library seemed to all be -1 and caused errors.

For these tasks, I used the BERT transformer. BERT (Bidirectional Encoder Representations from Transformers) was created at Google and is the application of a bidirectional transformer to language modelling. It reads entire sequences of text at once, as opposed to left to right. It is a very large transformer, with 110 million trainable parameters in its original form and 340 million trainable parameters in its large form.

BERT achieved very high metrics on natural language tasks, and it spawned a large amount of follow-on research. For that reason, we wanted to use BERT for some of our tasks.

Detailed Description of My Work

For this work, I loaded the data, tokenized and batched the data, defined model parameters, trained the model, and evaluated the model.

For a deeper look at my code, the MNLI and QNLI code is found in the repository.

Results

First, I trained MNLI and did validation on MNLI-M and MNLI-MM. Since the training was the same, both models share the same parameters. I ran into issues with CUDA memory, therefore I had to reduce training sample and epochs greatly. Due to this, I had very poor performance. My model did not outperform simply guessing the most common class.

Model: BERT **Epochs: 2** **LR: 0.0005** **Batch size: 8**
Accuracy: 35.45%

Model: BERT **Epochs: 2** **LR: 0.0005** **Batch size: 8**
Accuracy: 35.22%

Diagnostic would be included here, but I did not end up training it.

Next, I trained and did validation on QNLI. I got a much more respectable accuracy score here.

Model: BERT **Epochs 2:** **LR: 4.7e-5 (variable by partial epoch)** **Batch size: 4**
Accuracy: 79.736%

Summary and Conclusion

Due to BERT's size, it likely ran slower than slimmed down transformers like DistilBERT. The additional parameters of BERT made it slowly to run on our AWS instance with one GPU. Therefore, using a different transformer likely would have been better and produced better metrics.

Percentage of the Code Copied from the Internet

Around 75% of my code was taken from internet articles and other repositories. I then had to modify and put together the code to fit my use cases.

References

1. Horev, Rani. "Bert Explained: State of the Art Language Model for NLP." *Medium*, Towards Data Science, 17 Nov. 2018, https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270.

2. Lutkevich, Ben. "What Is Bert (Language Model) and How Does It Work?" *SearchEnterpriseAI*, TechTarget, 27 Jan. 2020, https://searchenterpriseai.techtarget.com/definition/BERT-language-model.

3. Verma, Dhruv. "Fine-tuning Pre-Trained Transformer Models for Sentence Entailment." *Towards Data Science,* 14 Jan. 2021. https://towardsdatascience.com/fine-tuning-pre-trained-transformer-models-for-sentence-entailment-d87caf9ec9db