Report

BERT

The language model called BERT (Bidirectional Encoder Representations from Transformers) was created by Google AI Language and is based on transformers. Presented in 2018, it has been pre-trained on an extensive text corpus to understand the contextual relationships among words in a phrase. Because BERT can comprehend the context of words in a phrase, it can do a variety of natural language processing tasks, including named entity identification, sentiment analysis, and question answering, with more accuracy and significance. It has become a well-liked tool in the natural language processing area and has demonstrated state-of-the-art performance on several benchmark datasets.
Many natural language processing (NLP) applications have found success using BERT, such as:

- Text classification
- Question answering
- Sentiment analysis
- Named entity recognition
- Language translation
- Chatbots
- Summarization
- Text generation
- Conversational AI
- Text entailment

BERT has shown to be especially useful for activities like question answering and language translation that call for an awareness of the context in which a word or phrase is employed. It has also been demonstrated to be useful in raising the accuracy of different NLP models.

Submissions

File1

We have executed the whole file for the first file, BERT QA Demo, which comprises Part 1 of the file, Implementing BERT model on COQA dataset.

Part 1: This section showed how the BERT model was applied to the COQA (Conversational Question Answering) dataset. Conversational questions and answers written by humans make up the COQA dataset. This solution uses a pre-trained BERT model, which was trained on a sizable corpus of text data.

Part2 : The Squad dataset was subjected to a refined BERT model.

This part used the SQuAD dataset to test the improved BERT model. Wikipedia articles with the related questions and answers are included in the SQuAD dataset. The SQuAD dataset was used to refine the BERT model. This enhances the model's ability to respond to inquiries in an organized manner. Tokenization, model training, assessment, and data preparation are all included in the implementation. Optimizing the model using the SQuAD dataset enhances its ability to respond to queries in a conversational manner. Tokenization, model training, assessment, and data preparation are all included in the implementation.

In summary, the BERT QA Demo shows how the BERT model can be applied to conversational and structured datasets for question-answering tasks, and how optimizing the model for particular datasets may enhance its overall performance.

File2

In Part3_Q2, Using the CoQA (Conversational Question Answering) dataset, we fine-tuned a pre-trained BERT (Bidirectional Encoder Representations from Transformers) model. BERT is a potent pre-trained language model that may be optimized for a variety of natural language processing applications, including text classification, sentiment analysis, and question answering. A pre-trained BERT model is fine-tuned by training it on a particular job, in this example, answering questions. With this strategy, we may make use of the BERT model's pre-existing knowledge and skills while tailoring it to the particular job at hand.

We started the pre-trained BERT model during the fine-tuning phase and put a second dense layer on top to categorize each question's response as "yes," "no," or "unknown." After that, we trained the model for five epochs, and the result was a 21.15% accuracy score. It's important to remember that if the model had been trained for more epochs, the accuracy score may have been greater. We must, however, strike a balance between training duration and attaining optimal performance because BERT model training requires a substantial amount of runtime and computing resources.

Overall, the effectiveness and adaptability of pre-trained language models in NLP tasks is demonstrated by our application of fine-tuning the BERT model on the CoQA dataset. On a variety of NLP tasks, we may get state-of-the-art performance by utilizing prior knowledge and fine-tuning for particular tasks.

File3

In 3rd file Part3_Q3&4, We employed the sentiment analysis of movie reviews collected from the 'Sentiment Analysis on Movie Reviews' Kaggle competition. The movie reviews in the 'Phrases' column, which are categorized as negative, slightly negative, neutral, somewhat positive, and positive and are represented by the numbers 0, 1, 2, 3, and 4, served as our training set. To answer the third issue, we applied a variety of natural language processing (NLP) techniques, including tokenization and encoding, to the movie review data in order to develop the BERT model. We set the BERT model with ignore_mismatched_sizes=True, utilizing the textattack/bert-base-uncased-imdb library.

The dataframe we have used in the notebook.

```
class_df = pd.read_csv('/content/drive/MyDrive/train.tsv', delimiter='\t', encoding='ISO-8859-1')
class_df # reading the data
```

	PhraseId	SentenceId	Phrase	Sentiment
0	1	1	A series of escapades demonstrating the adage ...	1
1	2	1	A series of escapades demonstrating the adage ...	2
2	3	1	A series	2
3	4	1	A	2
4	5	1	series	2
...	...	...	...	...
156055	156056	8544	Hearst 's	2
156056	156057	8544	forced avuncular chortles	1
156057	156058	8544	avuncular chortles	3
156058	156059	8544	avuncular	2
156059	156060	8544	chortles	2

156060 rows × 4 columns

Using the movie review data as training data, we were able to acquire an accuracy score of 21.83%. This accuracy score shows how well the machine can predict the sentiment label of a particular text review for a movie. The model does a better job at its task the higher the accuracy score.

```
#Data Accuracy without Finetuning
print(f'Accuracy for this data without finetuning BERT is {accuracy *100:.2f}%')
```

```
Accuracy for this data without finetuning BERT is 21.83%
```

We used the same dataset to apply the Finetuned Bert model in the same file.The process of fine-tuning a pre-trained BERT model entails extending its training on a novel assignment. Using the movie review dataset, we were able to refine the pre-trained BERT model for the sentiment analysis job.

By fine-tuning the pre-trained BERT model, we may utilize its acquired knowledge for the particular purpose of sentiment analysis. The model has previously learnt contextual word embeddings through extensive training on a vast quantity of text data. In order to do this, we load the pre-trained model first, and then we build a classification layer on top of the BERT model in order to forecast the input movie review's sentiment label. Two steps are involved in the fine-tuning process: the forward pass, which tokenizes, encodes, and runs the input phrase through the BERT model; and the backward pass, which computes the gradients and uses backpropagation to update the model's parameters. During training, the weights of the classification layer and the BERT model are updated by minimizing the loss between the predicted sentiment label and the true sentiment label using an optimizer such as Adam. We can utilize the refined BERT model to accurately predict the sentiment of fresh movie reviews once it has been trained. With the movie review dataset, we were able to obtain an accuracy score of 67.9% in this instance, which is a notable increase from the original result of 21.83% without fine-tuning.

```
100%|██████████| 3902/3902 [02:36<00:00, 24.98it/s]
Accuracy: 0.6795463283352556
Precision: 0.6770457694980494
Recall: 0.6795463283352556
F1-score: 0.6776426840134133
Confusion Matrix:
[[  758   581    73     3     1]
 [  680  3183  1592    69     3]
 [  149  1538 12503  1391    58]
 [   15   136  2065  3745   746]
 [    4     3    77   818  1021]]
```

Key Points

- The GPU runtime with high RAM was used to run the model on Colab Pro.