Machine Learning-2

Prabhas Reddy

April 2024

1 Overview

In the assignment, two variants of the pre-trained language model (LM) GPT-2 have been utilized: GPT-2 and GPT-2 Medium. Here is a breakdown of their sizes:

- GPT-2: 125.11 million parameters
- GPT-2 Medium: 356.62 million parameters

Both of these models are being fine-tuned using different approaches for the COLA dataset. The COLA dataset is designed for binary classification, specifically determining whether a given sentence is grammatically correct or not.

LoRA fine-tuning is used initially, followed by knowledge distillation to train a smaller RNN model using the fine-tuned GPT-2 or GPT-2 Medium model as the teacher.

2 Problem-1

2.1 LoRA

LoRA, which stands for Low-Rank Adaptation, is a popular technique to fine-tune LLMs more efficiently. Instead of adjusting all the parameters of a deep neural network, LoRA focuses on updating only a small set of low-rank matrices.

Specifically, LoRA separates pre-trained parameters by decomposing the weight matrix $X \in R^{m \times n}$ into U and V, where $U \in R^{m \times r}$ and $V \in R^{r \times n}$.

This reduces the parameter space from $m \times n$ to $(m \times r) + (r \times n)$, enabling faster training and lower computational cost compared to full fine-tuning. Then these values will be merged with the pretrained weights of the Llama model to obtain one finetuned model.

3 Fine Tuning GPT-2

During the fine-tuning process of the GPT-2 variants, specific layers are frozen while lower-rank matrices are injected for fine-tuning. After fine-tuning, these modified weights are merged with the original pretrained parameters. Here are the layers involved in this process:

1. Casual attention
2. Casual Projection
3. Casual Fully connected layer in MLP
4. Casual Projection in MLP

For each of these layers, lower-rank matrices are used during fine-tuning, and the resulting weights are combined with the pretrained parameters to enhance the model's performance on the COLA dataset.

This table summarizes the original number of parameters and the number of trainable parameters after LoRA fine-tuning for both GPT-2 and GPT-2 Medium variants.

Model Variant	**Original Params (M)**	**Trainable Params (M)**
GPT-2	125.11	0.71
GPT-2 Medium	356.62	1.90

Table 1: Comparison of Parameters Before and After LoRA Fine-Tuning

3.1 LoRA Linear Class

The LoRA Linear class is composed of two lower-rank matrices designed for LoRA fine-tuning. These matrices are initialized using Xavier Uniform Initilization for optimal initialization and are further enhanced with a dropout of 0.2 for regularization.

During the forward pass, the input matrix undergoes multiplication with both lower-dimensional matrices and is subsequently outputted. Additionally, an alpha parameter is introduced to scale the output generated from this LoRA Linear class.

3.2 Training

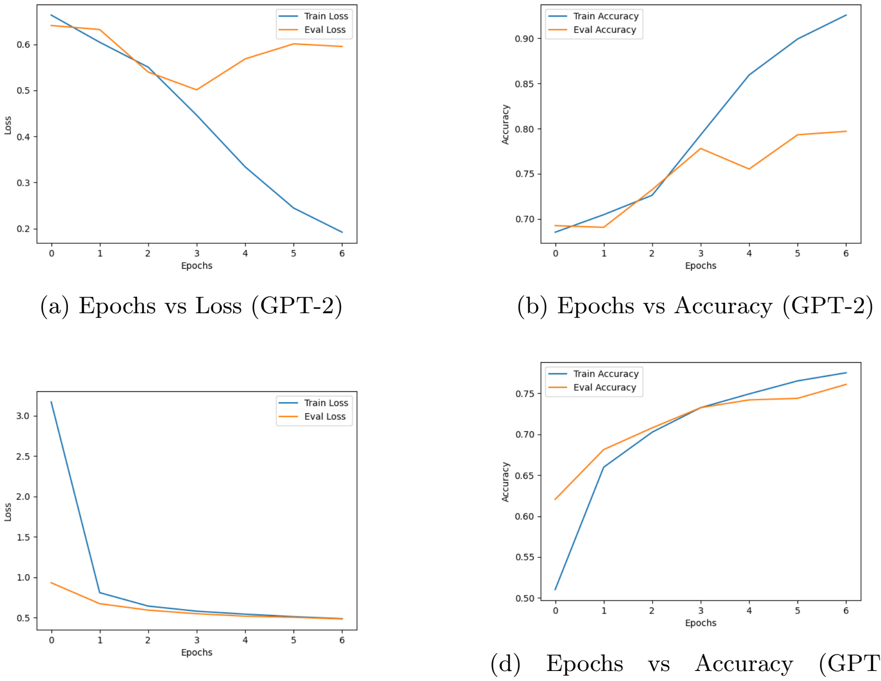
The following table outlines the configurations used for fine-tuning using LoRA:

Model Variant	**Learning Rate**	**Batch Size**	**Epochs**	**LoRA Rank**	**Alpha**
GPT-2	0.001	256	7	4	8
GPT-2 Medium	2e-5	256	7	4	8

A lower learning rate is used for GPT-2 Medium due to its larger size. These configurations were selected after a thorough hyperparameter search and have been observed to work effectively.

3.3 Results

These are graphs obtained for epochs vs loss and epochs vs accuracy for 2 variants of GPT2 while fine tuning.



(a) Epochs vs Loss (GPT-2)

(b) Epochs vs Accuracy (GPT-2)

(c) Epochs vs Loss (GPT-2 Medium)

(d) Epochs vs Accuracy (GPT-2 Medium)

Figure 1: Graphs for epochs vs loss and epochs vs accuracy for GPT-2 variants

The best obtained accuracies for both variants are mentioned below in following table.

Model Variant	**Best Accuracy (%)**
GPT-2	80.06
GPT-2 Medium	83.25

Table 2: Best Obtained Accuracies for GPT-2 Variants

4 Problem -2

4.1 Knowledge Distillation

Knowledge distillation refers to the process of transferring knowledge from a large unwieldy model or set of models to a single smaller model that can be practically deployed under real-world constraints. In a neural network, knowledge typically refers to the learned weights and biases. It is proven to be an effective method to transfer knowledge from a bigger model to a smaller model, resulting in building a smaller model with similar capabilities of the larger model.Here bigger models is termed as Teacher Model and smaller one as Student Model.

4.2 Teacher and Student Models Used

In this problem, trained variants of GPT-2 that are saved from Problem-1 after fine-tuning are used as teacher models to transfer knowledge to a smaller RNN for the COLA dataset binary classification.

The Student model architecture used is as follows:

- Embedding Layer (`self.embed`): This layer transforms the input word indices into dense vectors of fixed size. In this case, the input size is 50257 (the size of the GPT-2 vocabulary), and the output size is 768 (the size of the GPT-2 embeddings). The output of this layer for each word is a 768-dimensional vector.
- GRU Layer (`self.rnn`): This is the recurrent layer of the network. It takes the 768-dimensional word embeddings as input and outputs a sequence of hidden states of the same dimensionality (768). The number of layers in the GRU is 1, meaning it's a single-layer RNN. GRU is used because it has less problem with vanishing gradients compared to RNN.
- Linear Layer (`self.fc`): This is a fully connected layer that maps the 768-dimensional hidden states to a 2-dimensional output. This is used for binary classification, where each dimension corresponds to the score of a class.

4.3 Training

In knowledge distillation, the Teacher model's weights remain unchanged, focusing solely on updating the Student model's weights. The loss function is designed to optimize two key aspects: improving the Student model's classification accuracy and ensuring its logits closely match those of the Teacher model. To achieve this, a hyperparameter called 'fraction' is introduced, allowing for a balanced adjustment between these objectives. Mathematically, the loss components are defined as follows:

- **Student_loss**: Represents the cross-entropy loss of the Student model.

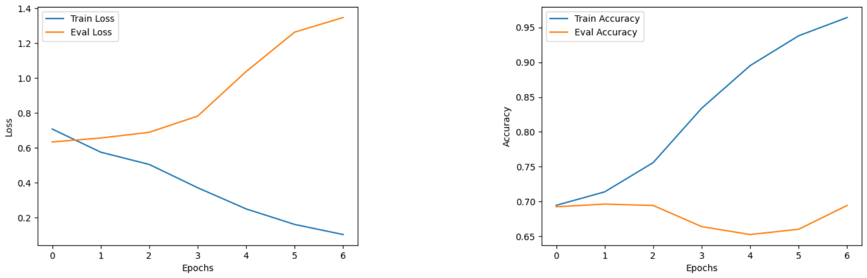
- **Distillation_loss**: Indicates the difference between the logits of the Teacher and Student models.
- **Final_loss**: Combines the student_loss and distillation_loss based on the fraction parameter:

$$Final\_loss = fraction \times Student\_loss + (1 - fraction) \times Distillation\_loss$$

Here, the fraction parameter ranges from 0 to 1, offering a flexible mechanism to balance the influence of each loss component. By optimizing Final_loss, the gradients of the Student model are adjusted to converge towards a performance level similar to that of the Teacher Model.

4.4 Results

These are graphs obtained for epochs vs loss and epochs vs accuracy while Training the Student Model alone.



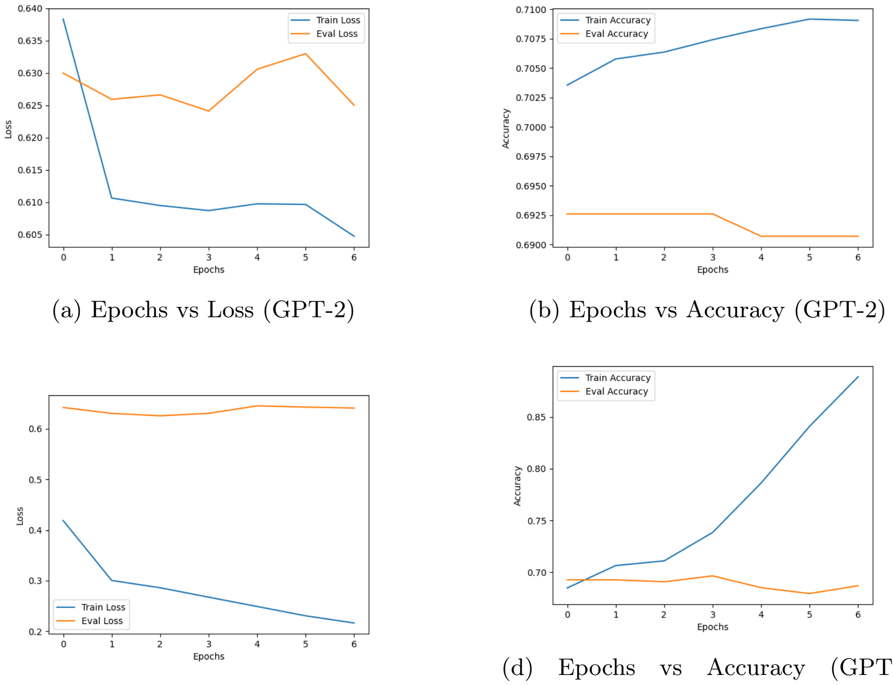
(a) Epochs vs Loss (RNN)

(b) Epochs vs Accuracy (RNN)

Figure 2: Graphs for epochs vs loss and epochs vs accuracy on Student Model

These are graphs obtained for epochs vs loss and epochs vs accuracy while Training the Student Model from different variants as Teacher Models with Knowledge Distillation.

The best obtained accuracy for Student Model after training is 69.46 percent. The best obtained accuracies for both variants after Knowledge Distillation are mentioned below in following table.



(a) Epochs vs Loss (GPT-2)

(b) Epochs vs Accuracy (GPT-2)

(c) Epochs vs Loss (GPT-2 Medium)

(d) Epochs vs Accuracy (GPT-2 Medium)

Figure 3: Graphs for epochs vs loss and epochs vs accuracy on Student Model with Knowledge Distillation with different Teacher Models.

Student Model	**Teacher Model**	**Best Accuracy (%)**
RNN	GPT-2	69.85
RNN	GPT-2 Medium	70.26

Table 3: Best Obtained Accuracies for RNN after Knowledge Distillation

5 Conclusion

LoRA fine-tuning and Knowledge Distillation are highly effective methods for fine-tuning large Language Models (LLMs) to specific tasks with remarkable efficiency and minimal loss in performance. These techniques enable the adaptation of pre-trained models like GPT-2 and GPT-2 Medium to new tasks such as binary classification, demonstrating their versatility and utility in NLP tasks. Additionally, LoRA fine-tuning allows for targeted updates to specific layers, reducing computational costs and training time. Knowledge Distillation further enhances model deployment by transferring knowledge from complex teacher models to more lightweight student models, striking a balance between accuracy and resource efficiency.