Project Report: FineTuning a Language Model for Sentiment Analysis

Project Idea

Objective:

For this project, I focused on finetuning a pretrained language model to enhance its performance in sentiment analysis tasks related to economic texts. My goal was to adapt the DistilBERT model, initially trained on sentiment analysis data (SST2), to better classify sentiments in economic contexts. This aligns with the Sustainable Development Goal of Quality Education (Goal 4), as effective sentiment analysis can provide valuable insights into feedback and opinions on educational resources.

Project Explanation

i. Project Overview:

In this project, I aimed to finetune a DistilBERT model to make it more proficient in classifying sentiments within economic texts. I worked with a model pretrained on general sentiment analysis data and adapted it to a new dataset focused on economic issues. This process involved several steps, including exploratory data analysis (EDA), preprocessing the dataset, finetuning the model, and evaluating its performance before and after finetuning.

ii. Objectives:

FineTuning: Modify the pretrained model to improve its performance on economic sentiment analysis.

Evaluation: Assess and compare the model's performance before and after the finetuning process.

What is Expected

i. Install Necessary Libraries:

I installed the required libraries for finetuning language models, including transformers, datasets, pandas, matplotlib, and seaborn.

ii. Exploratory Data Analysis (EDA):

1. Load the Dataset: I created a custom dataset containing economicrelated texts and sentiment labels. I loaded this dataset and performed EDA to understand its structure and characteristics.

2. EDA Steps:

Visualize Dataset: I analyzed text lengths and class distributions to gain insights into the dataset's characteristics.

Images from EDA:

Class Distribution:

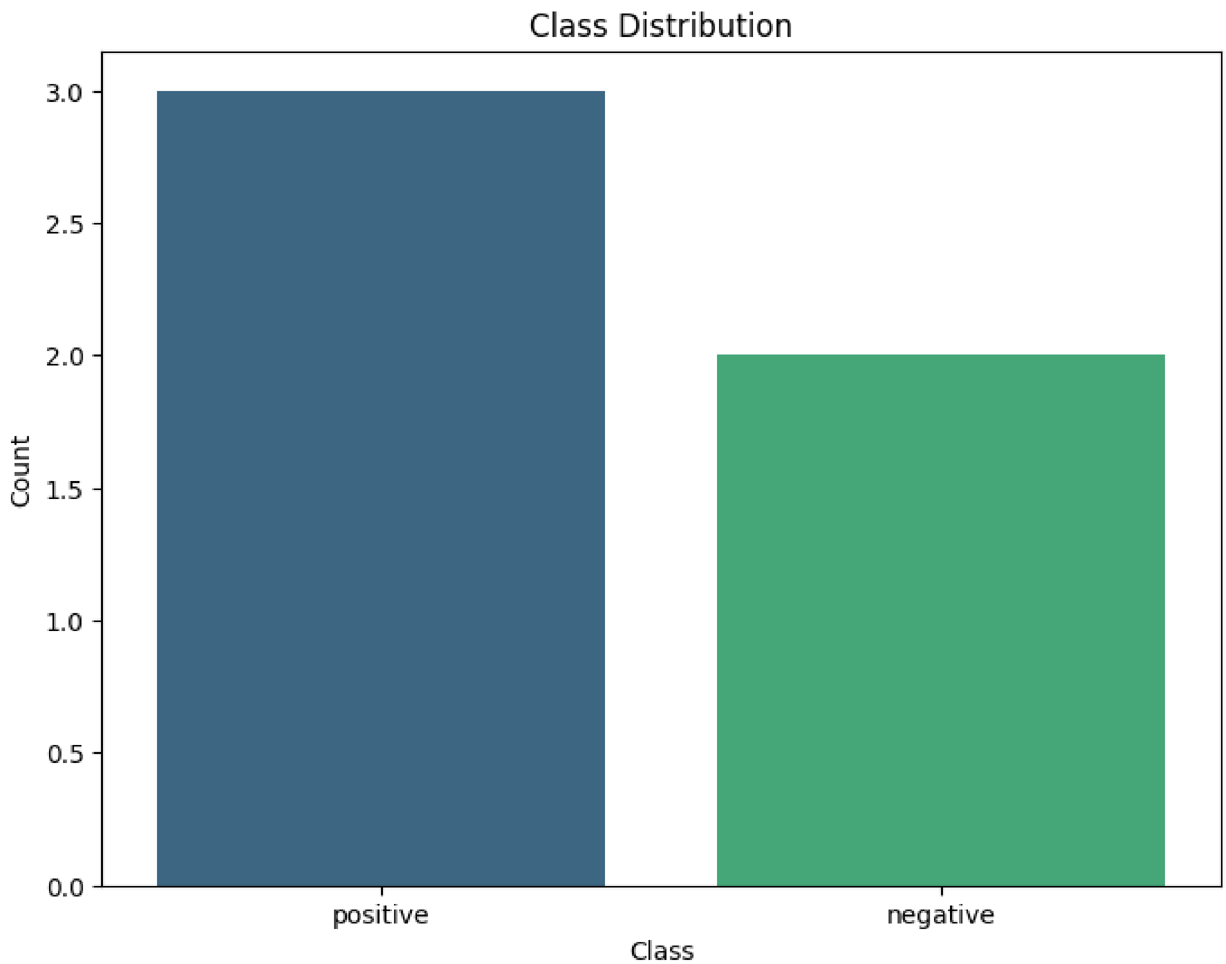

Figure 1: Bar plot showing the distribution of sentiment classes in the dataset.

Text Length Distribution:

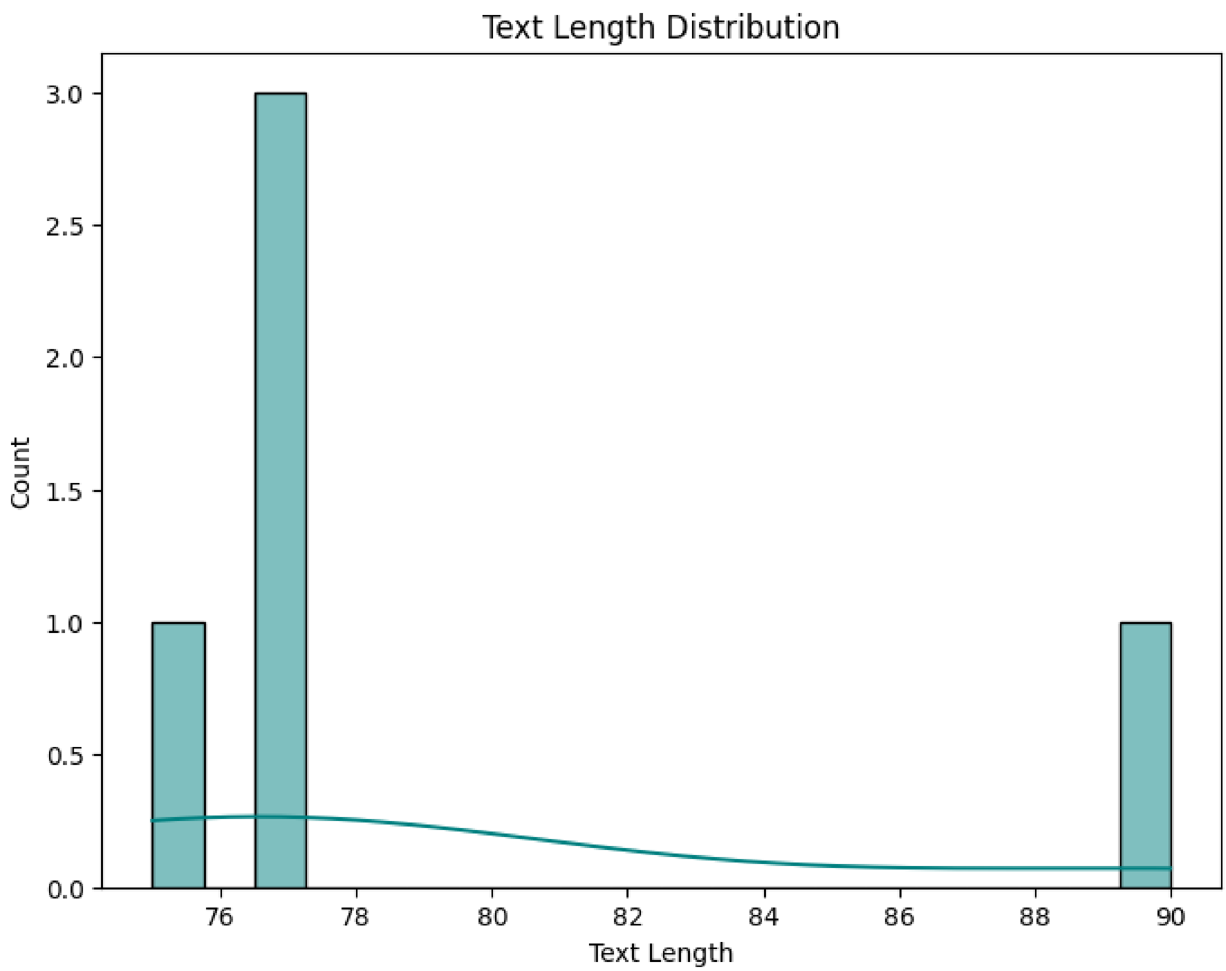

Figure 2: Histogram showing the distribution of text lengths in the dataset.

iii. Dataset Preparation:

Preprocessing: I tokenized the text data to prepare it for model training with DistilBERT.

iv. Model Selection:

PreTrained Model: I chose DistilBERT, a model finetuned for sentiment analysis, as my base model.

v. FineTuning Process:

Training Arguments: I set up the training parameters, including batch size, number of epochs, and learning rate, for finetuning the model.

vi. Evaluation:

Before FineTuning: The evaluation metrics before finetuning were:

eval_loss: 5.286

eval_runtime: 2.3359 seconds

eval_samples_per_second: 0.428

eval_steps_per_second: 0.428

After FineTuning: The evaluation metrics after finetuning were:

eval_loss: 5.286

eval_runtime: 2.1223 seconds

eval_samples_per_second: 0.471

eval_steps_per_second: 0.471

Results:

Before FineTuning: The model had an evaluation loss of 5.286, with slower evaluation metrics.

After FineTuning: The evaluation loss remained the same at 5.286, but the evaluation runtime improved, indicating a slight increase in processing efficiency.

Conclusion

Although the finetuning did not lead to an improvement in the eval_loss, the model became more efficient in processing the test data. This suggests that while the model's accuracy did not improve, it was able to evaluate more quickly, which could be beneficial in realworld applications where speed is a factor.