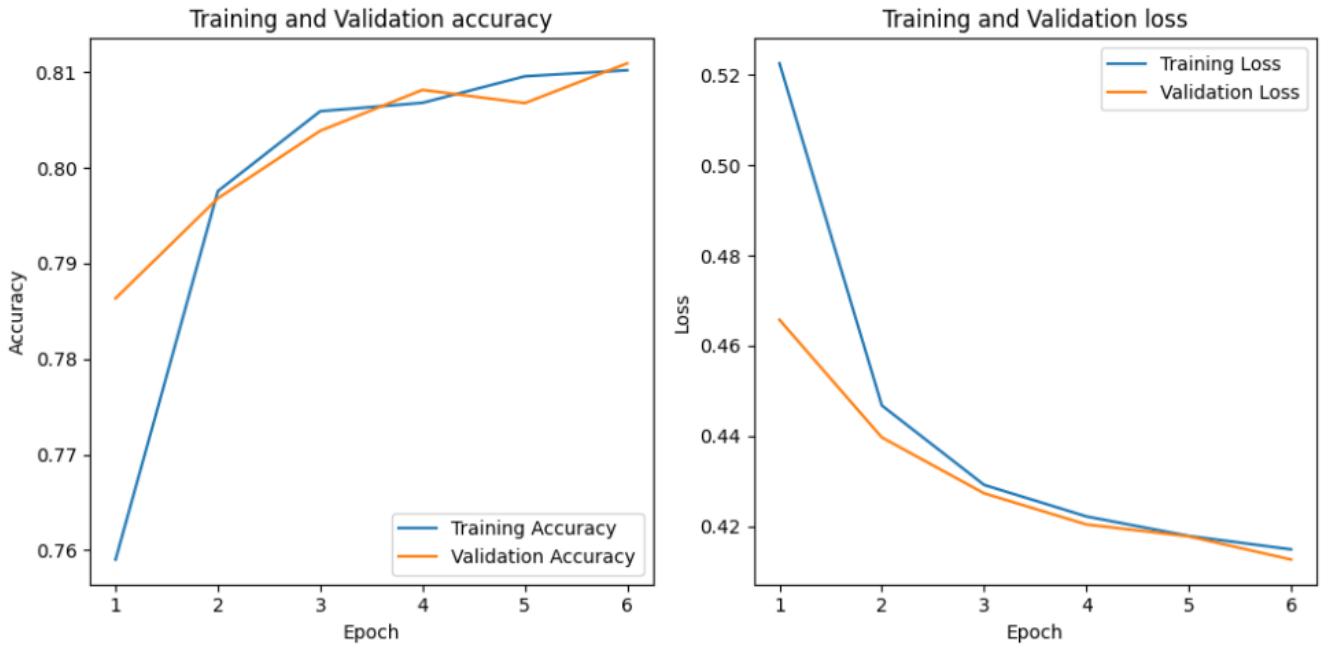
AI Final

Link to Jupyter Notebook: https://github.com/John-A-Aydin/AI-Final/blob/main/Final.ipynb

> **Important**
>
> I tried a few variations of the model architectures shown in the links that were provided, but ended up choosing the simple one seen in the notebook, due to some weird typecasting errors I was getting from tensorflow and my GPU running out of memory (It's the lowest end NVIDIA gpu from 8 years ago).

Question 1:



The graphs show that the model learns the most in the first 2-3 epochs and that there is minimal over-fitting since the curves for loss and accuracy are both better in validation than training.

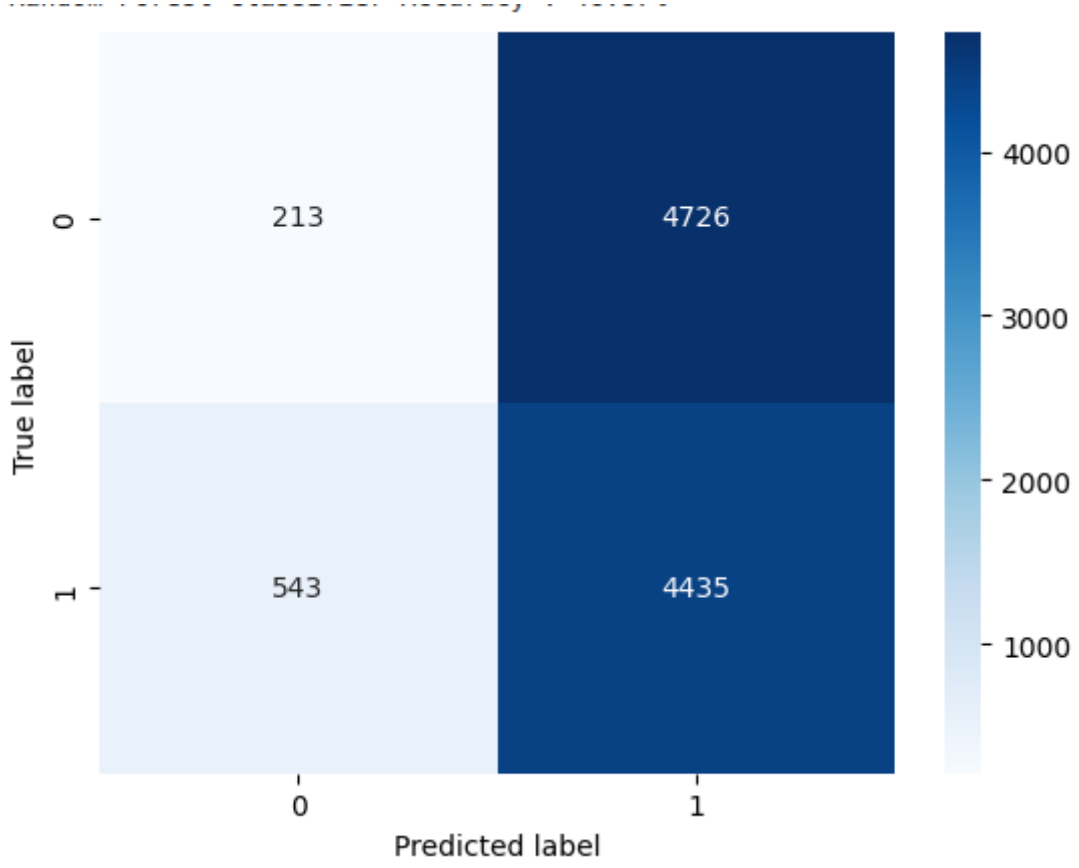
This may indicate that the model is under-fitting, but the results are so close that they are negligible.

Question 2:

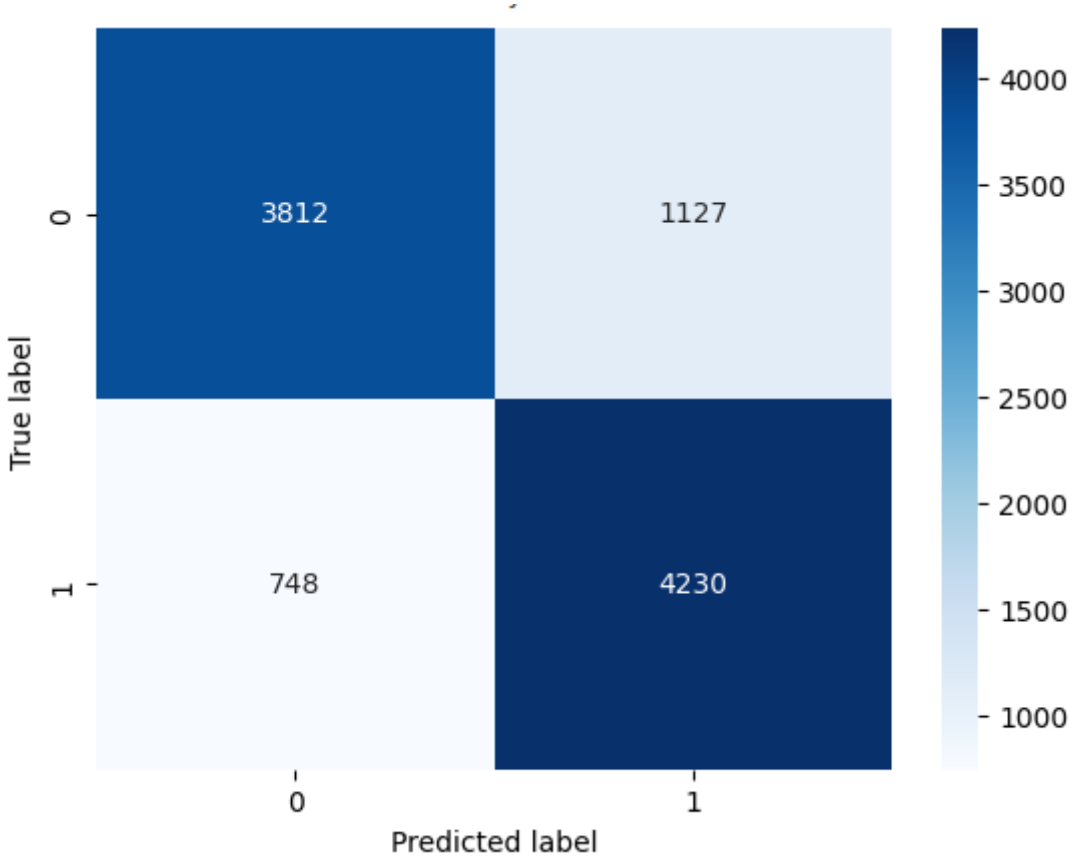
The fine-tuned model performs slightly worse than the classical ML model. Transformers are better at contextualizing sequential data, but that advantage is somewhat limited in this exercise since we discard non-descriptive words. This makes it easier for the classical model to pick up on patterns while avoiding noise, but takes away the fine-tuned model's main advantage.

Question 3:

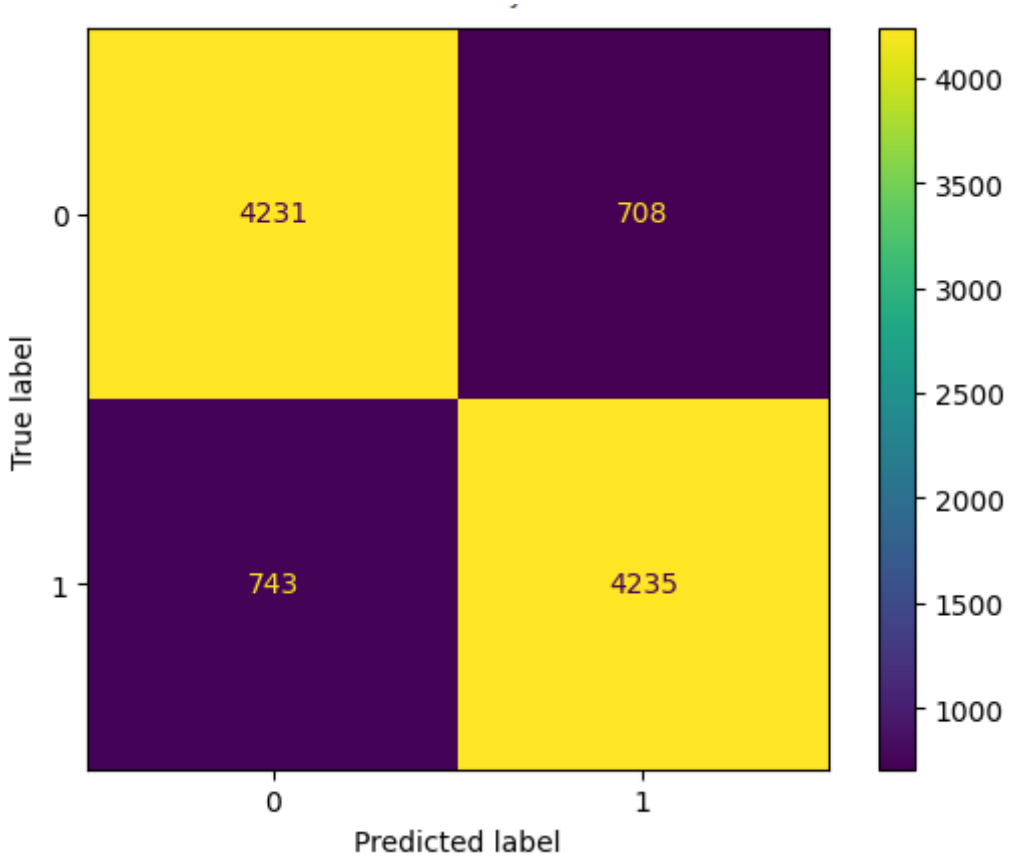
Base DistilBERT



Fine-tuned DistilBERT



Random Forest Classifier



The only patterns that can be seen in the confusion matrices are the bias of base DistilBERT towards classifying reviews as positive and the fine-tuned DistilBERT's slight adherence to that bias.

The RF model seems to be very unbiased with only negligible differences between misclassifications.

Question 4:

Exposing the fine-tuned model to the data set allows it to pick up on patterns that are unique to the application. Movie reviews may express negativity using different words or structure than something like restaurant reviews, or just written text in general.

Fine tuning allows us take a generalist model and hone in on one domain with specific conditions, creating better results.

> **Side note:**
>
> I'm not unsure if I ran the base model correctly given the results.
> I saw similar results even with more layers.

Question 5:

If the training times and text processing remained the same, I would use the Random Forest Classifier. Although the training time was significantly faster, this does not influence my decision since training time is a one time cost. The real benefit is the accuracy and performance of predictions. The RF model showed better accuracy and less bias than the fine-tuned model.

The RF model also ran on a CPU with comparable performance to the fine-tuned model which was running on a GPU. The cost alone of GPU compute would drive me to choose RF.

If the dataset was more dependent on context, like customer sentiment in a support chat, I would explore using a fine-tuned model. In this application, the text data wouldn't have as many dead giveaways of negativity which would hurt the performance of a classical model.

The cost of running this model would be a factor, but it would all depend on how business-critical the application is.

Classification Reports

Model	Precision (negative)	Recall (negative)	F1-Score (negative)	Precision (positive)	Recall (positive)	F1-Score (positive)
DistilBERT (base)	0.28	0.04	0.07	0.48	0.89	0.63
DistilBERT (fine-tuned)	0.84	0.77	0.80	0.79	0.85	0.82
Random Forest Classifier	0.85	0.86	0.85	0.86	0.85	0.85

Precision

The base DistilBERT model showed lower precision than choosing at random so it was a complete failure. The fine-tuned DistilBERT model showed comparable precision to the Random Forest Classifier (RF), but was slightly lower with its predictions for positive reviews.

Recall

The base DistilBERT model scored very low in it's recall rate for negative reviews, which is expected since it classified almost all of the reviews as positive. It's performance with positive reviews looks good on paper, but since the model almost elusively predicted that reviews were positive, this metric means almost nothing. A model that predicted every single review to be positive would receive a recall of 1.00 despite being incredibly flawed

The fine-tuned DistilBERT model had a slightly lower recall rate with negative reviews than RF, but a similar rate to RF for positive reviews. This finding shows that the RF model would be better at reliably identifying negative reviews than the fine-tuned model.

F1-Score

The F1-Scores of the models rank in the following order: base DistilBERT, fine-tuned DistilBERT, RF. Overall, RF was better in almost every metric, but the fine-tuned model was a close second.

Time Complexity

Random Forest vs. Base DistilBERT

The base DistilBERT model took roughly the same amount of time to get running as the random forest classifier, but produced very bad results. There is no advantage to using the LLM without fine-tuning.

Random Forest vs. Fine-tuned DistilBERT

The random forest classifier (RF) performed better than the fine-tuned DistilBERT model despite only taking roughly 2 minutes to train vs the 15+ minutes (4 hrs w/o GPU) of the fine-tuned DistilBERT model. The random forest classifier also showed similar prediction times on a CPU to that of the fine-tuned model running on a GPU. This shows that the RF model takes less resources.

The Random Forest classifier is by far the most efficient with both time and resources.