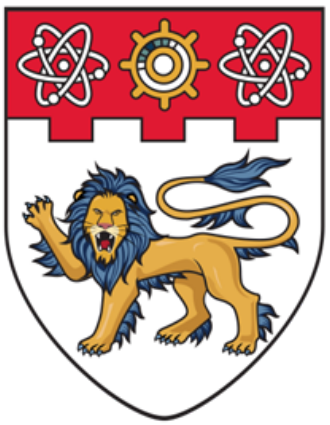


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

SC4001 Neural Networks

Names	Lim Jia Earn (U2122747A) Tan Wei Herng (U2122787D) JOSEPH FULL NAME (U2122493C)
Project Idea	C2

Table Of Contents

1. Introduction

In the evolving field of Natural Language Processing (NLP), the quest for context-aware and accurate models has ushered in new possibilities for text sentiment analysis (TSA). TSA enables machines to bridge the gap between human language and binary systems in areas such as language translation and social media monitoring. The importance of TSA is only expected to continue growing as the volume of digital data continues expanding.

2. Our Objective

In the realm of TSA, there are various formidable NLP models which have the capabilities to be able to comprehend context and dependencies in human language, a bedrock of sentiment analysis.

Our group's objective is to assess performance disparity among 4 popular models.

1) **CNN :** A convolutional neural network (CNN) adept at processing text or image data but faces challenges in capturing long range dependencies in text.
2) **BERT :** A popular attention model which can learn text sequences bidirectionally, giving it an edge to have a deeper sense of language context.
3) **RoBERTa :** An optimised variant of **BERT** where it went through optimisations to its pre-training process for a robust language understanding.
4) **DistilBERT :** A lightweight version of **BERT** architecture serving as another option where computation resources are a concern while distilling the essential features of its predecessor.

Our evaluation of the 4 models' performance will be through key metrics such as loss, accuracy, training and testing time which will shed light on their respective strengths and weaknesses.

3. Data Exploration and Pre-processing

All Codes for diagrams below are in SC4001_DataExploration.ipynb

With our extensive dataset comprising over 70 000 data rows, the data exploration stage is essential to guide our selection of pre-processing steps.

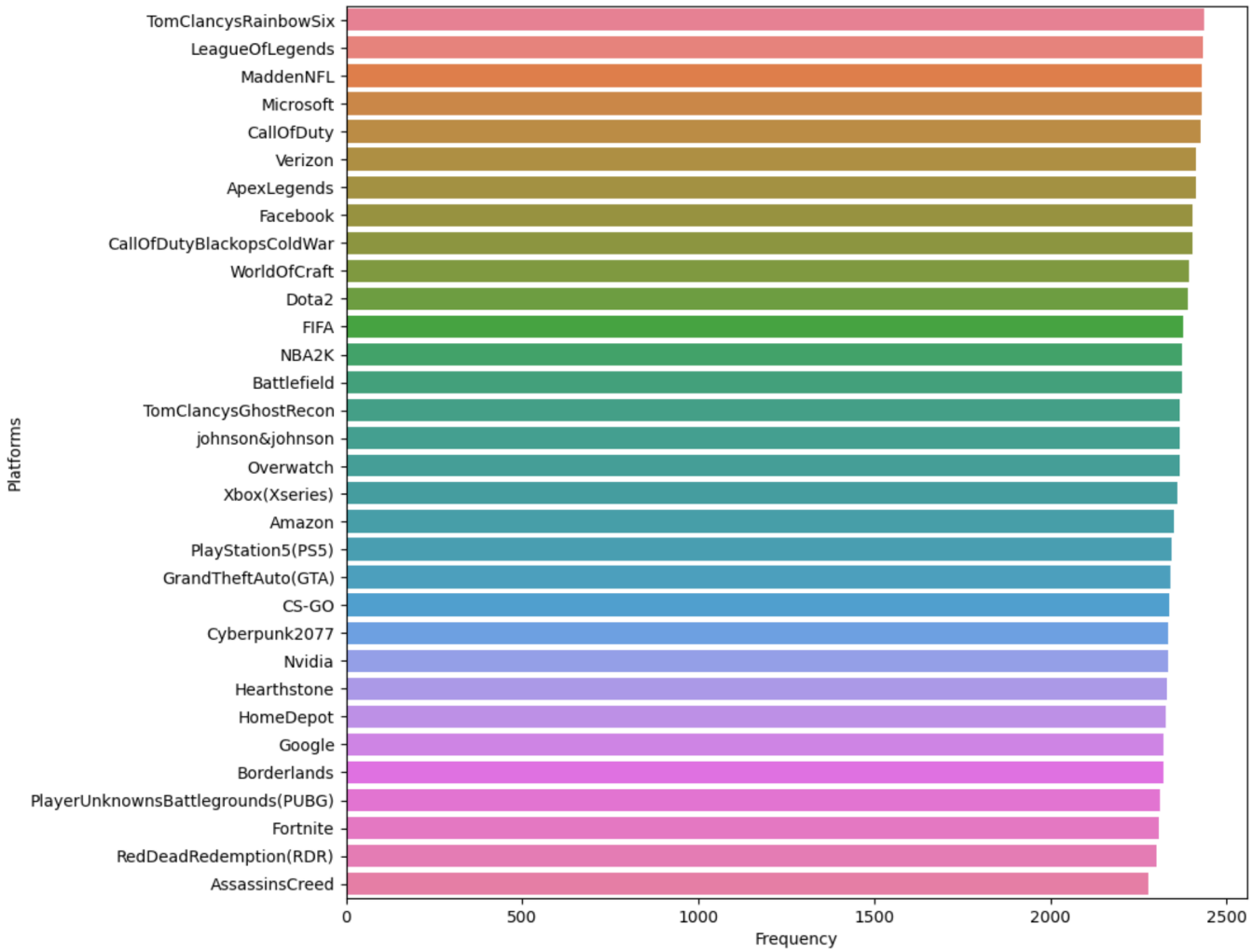


Figure 1

From Figure 1, our dataset encompasses social media platforms such as Facebook and various gaming platforms such as Xbox. This diversity introduces the possibility of special characters and emojis which could hinder the learning performance of our models which we will first remove.

To further optimise the learning performance of our model, we will also remove stop words. These words are abundant in our text data but do not provide meaningful information for our models. Removing these stopwords can reduce the data size which results in faster computations during our training and testing phase (wisdomml, 2022).

In order to enhance the efficiency of our training process, we convert sentences to lowercase to avoid unnecessary training of vectors. One drawback of this approach is that capitalization, which could potentially help predict strong emotions, is disregarded. Nevertheless, our model is designed to predict the overall nature of the text (e.g., Positive, Negative, Neutral) rather than quantify the intensity of emotions (e.g., Disappointment and Anger).

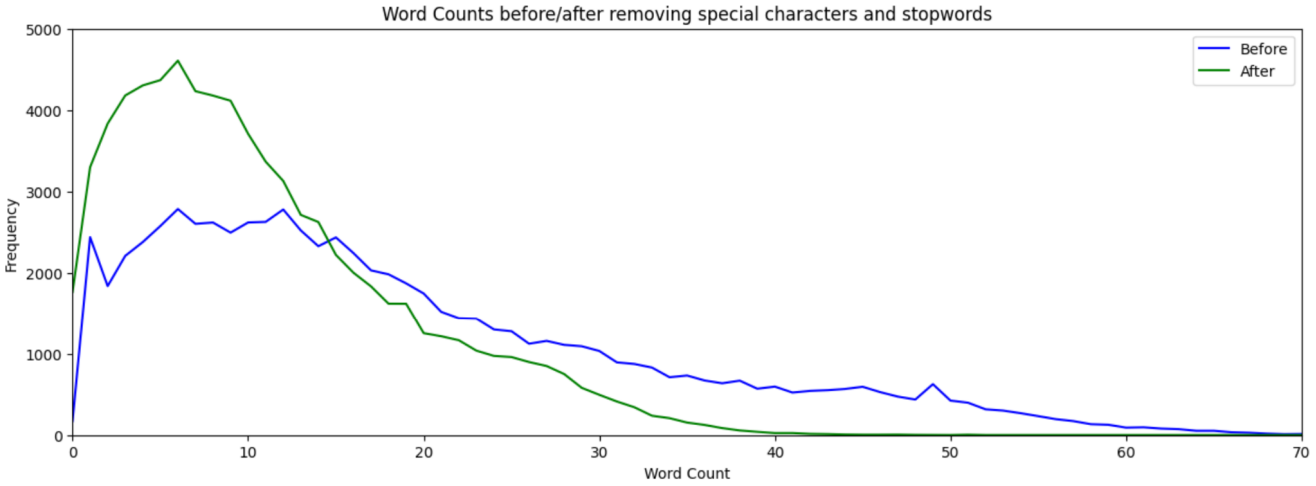


Figure 2

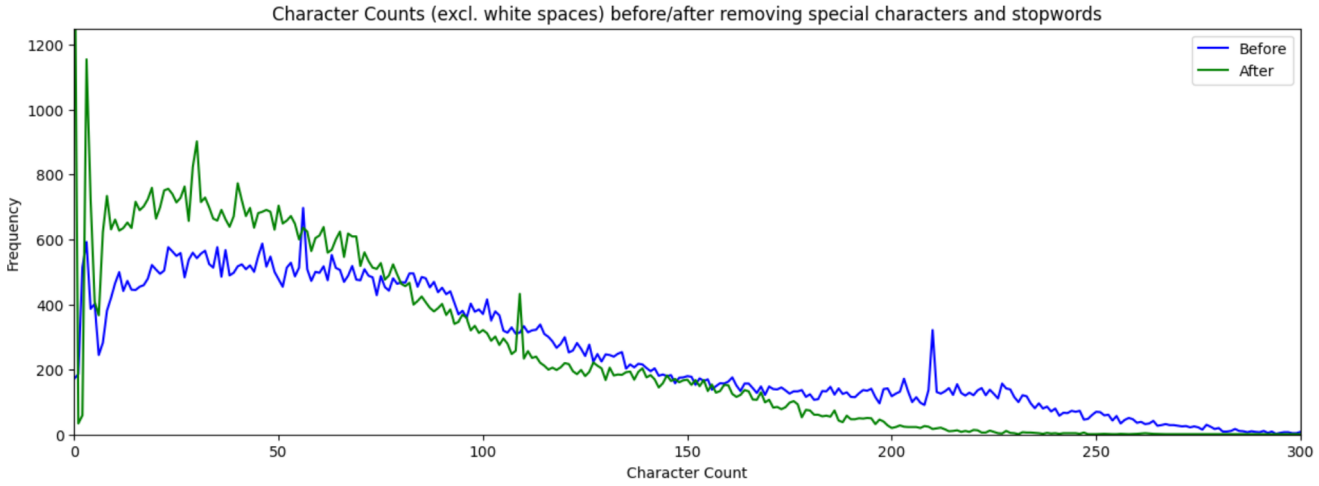


Figure 3

We can see a noticeable left translation of the distribution for both character and word counts as depicted in Figure 2 and 3 after the removal of special characters and stopwords. This streamlined distribution will provide for a more targeted and efficient analysis in our subsequent model training phase.

Lastly, we perform label mapping for the original labels to numeric values to facilitate our training and evaluation phase. We mapped “Neutral”/”Irrelevant”, “Positive”, “Negative” to 0, 1, 2 respectively. For readability of the report, we have used the original labels for our figure below.

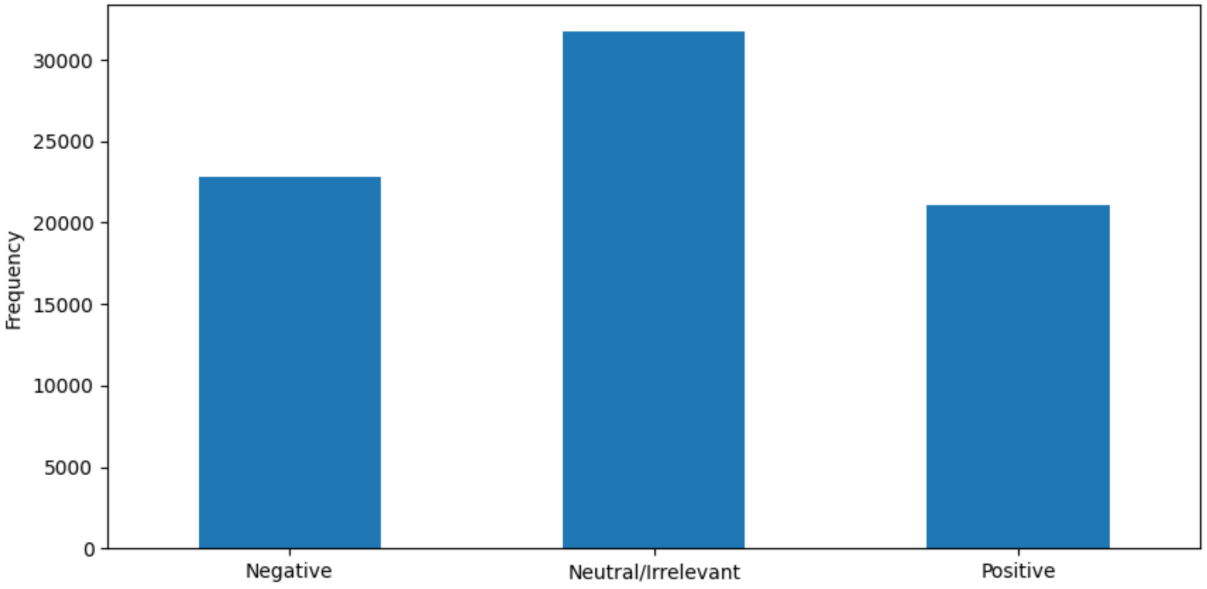


Figure 4

The data are reasonably well distributed between the 3 classes, so our group decided not to attempt the removal of any data rows just to perfectly balance our distribution.

To end off our data pre-processing journey, we generated a word cloud for a vibrant observation of our dataset. The size of the words correlates to its frequency in our textual data.

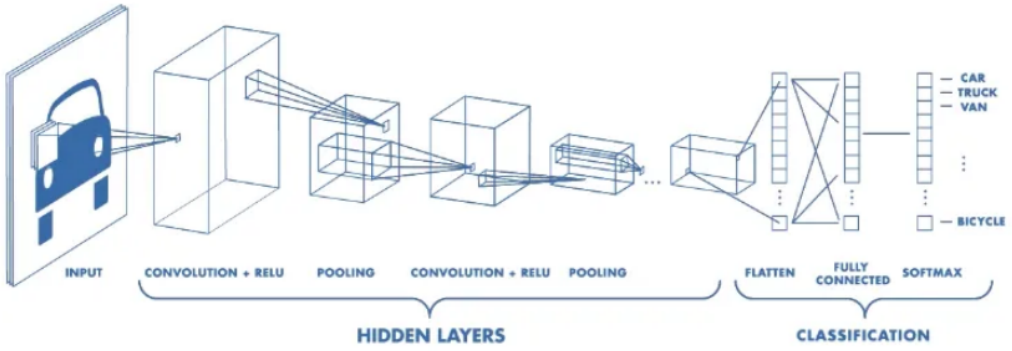
Figure 5

4. Models Training Results

With the completion of our data pre-processing and cleaning, we will move on to model training and testing phase to uncover the comparative strengths and efficiency of the models. For consistency of our time measurements, all of our 4 models are trained on Google Colab and using V100 GPU.

4.1 : CNN (Convolutional Neural Network)

CNN is a class of deep neural networks, popularly used in applications for analysing visual imagery. Its hidden layer consists of a convolution layer followed by a pooling layer and finally a softmax layer which will return the classifications. CNN can also be used for sentimental analysis on text data and classify the text data into different classes, however it is expected to struggle with the sequential structure of text data and thus will have limited understanding of the context of textual sentences.



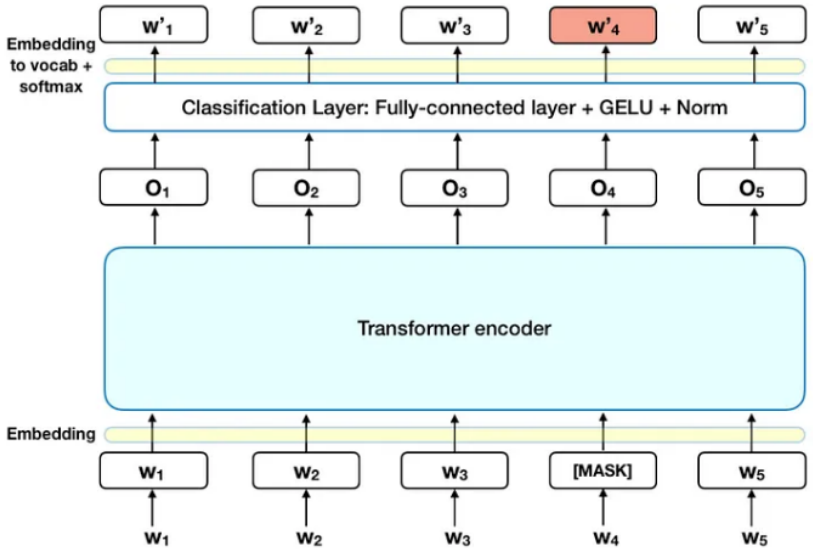
Below is the accuracy for the CNN model, with the validation accuracy around 41.9% for 3 epochs which proves the CNN model is not very effective for classifying sentences.

All Codes are in SC4001_CNN.ipynb

```
Epoch 1/3
821/821 [==============================] - 27s 33ms/step - loss: 0.2349 - accuracy: 0.3875 - val_loss: 0.2252 - val_accuracy: 0.4187
Epoch 2/3
821/821 [==============================] - 22s 27ms/step - loss: 0.2214 - accuracy: 0.4195 - val_loss: 0.2193 - val_accuracy: 0.4187
Epoch 3/3
821/821 [==============================] - 24s 29ms/step - loss: 0.2187 - accuracy: 0.4195 - val_loss: 0.2184 - val_accuracy: 0.4187
```

4.2 : BERT Architecture

BERT is a game-changer in NLP, thanks to its bidirectional understanding of text data. Its usage of attention mechanisms to grasp relationships make it formidable for tasks like sentiment analysis. BERT's core involves transformers that examine word correlations in a sentence. It has two parts: an encoder for input and a decoder for predictions. Unlike models reading sequentially, BERT processes the entire sequence bidirectionally, enhancing its effectiveness.This characteristic allows the model to learn the context of a word based on all of its surroundings.



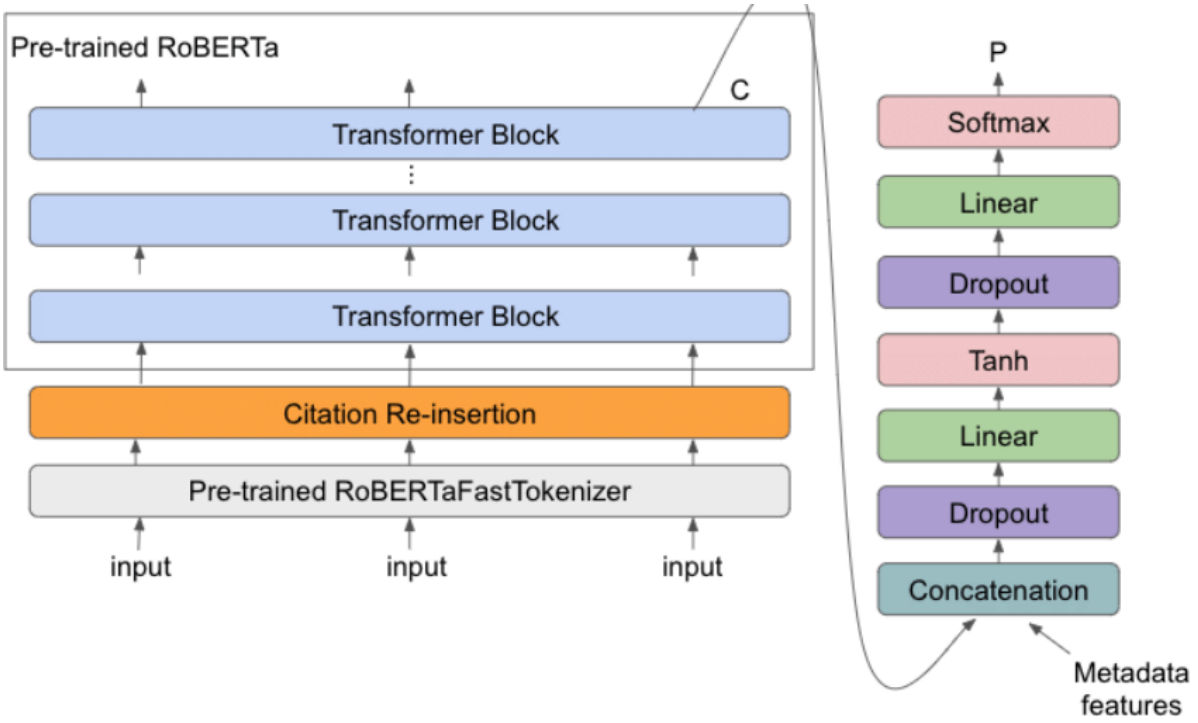
Below are the performance benchmark for our model BERT.

All Codes are in SC4001_BERT.ipynb

epoch	Training Loss	Valid. Loss	Valid. Accur.	Training Time	Validation Time	F1 Score
1	0.847278	0.732257	0.681095	0:09:34	0:01:03	0.673833
2	0.617353	0.588214	0.759772	0:09:31	0:01:03	0.760627
3	0.500595	0.553939	0.778029	0:09:31	0:01:03	0.778150

4.3 : RoBERTa

Building upon the foundation of BERT, RoBERTa refined the pre-training process for an enhanced language understanding.RoBERTa is a reimplementation of BERT with some modifications to the key hyperparameters and tiny embedding tweaks, along with a setup for RoBERTa pre-trained models.(Sharma,2022)



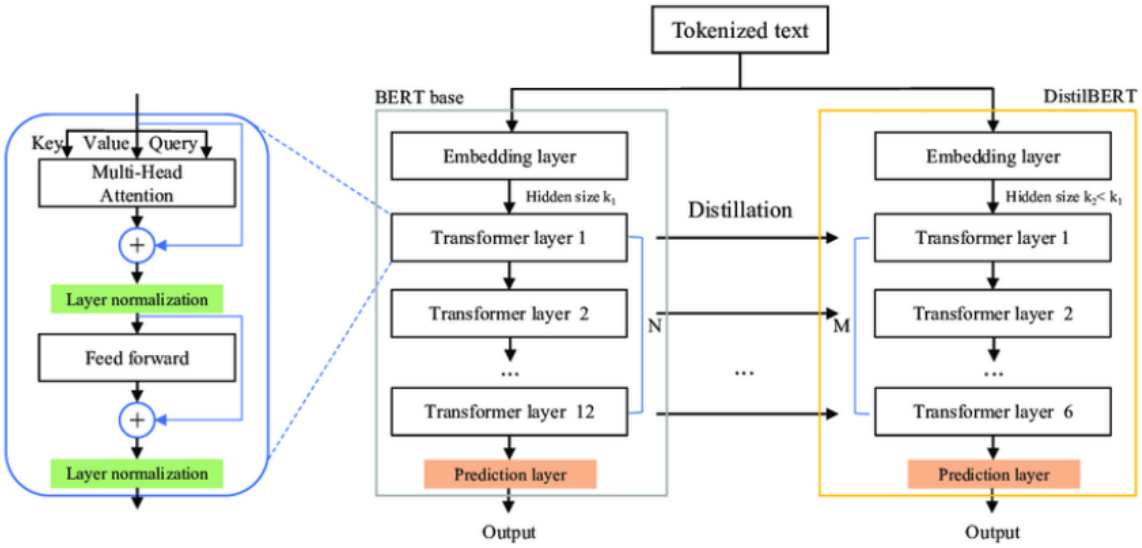
Our model took a total of about 30 mins to train. When used on our test dataset, the model achieved an accuracy of 86% and a F1 score of 86%. Below are the benchmarks of our model performance.

All Codes are in SC4001_roBERTa.ipynb

epoch	Training Loss	Valid. Loss	Valid. Accur.	Training Time	Validation Time	F1 Score
1	0.691924	0.564279	0.766808	0:09:37	0:01:01	0.766380
2	0.434418	0.416325	0.843672	0:09:35	0:01:01	0.843970
3	0.282314	0.385577	0.865993	0:09:35	0:01:01	0.865966

4.4 : DistilBERT

DistilBert is a distilled form of the BERT model where it is 40% smaller than the normal BERT model via knowledge distillation while retaining 97% of its language abilities and being 60% faster due to being smaller. A triple loss is introduced by combining language modelling, distillation, and cosine-distance losses to take advantage of the inductive biases learned by larger models during pre-training.DistilBERT is a compact, faster, and lighter model that is cheaper to pre-train and can easily be used for on-device applications (Sharma, 2022).
We hope to achieve significant results with a less computationally expensive model that can still compete with the large scale models.



Our model took a total of about 15 mins to train. When used on our test set, the accuracy was roughly 85%.and f1 score was 85%. Below are the benchmarks of our model performance.

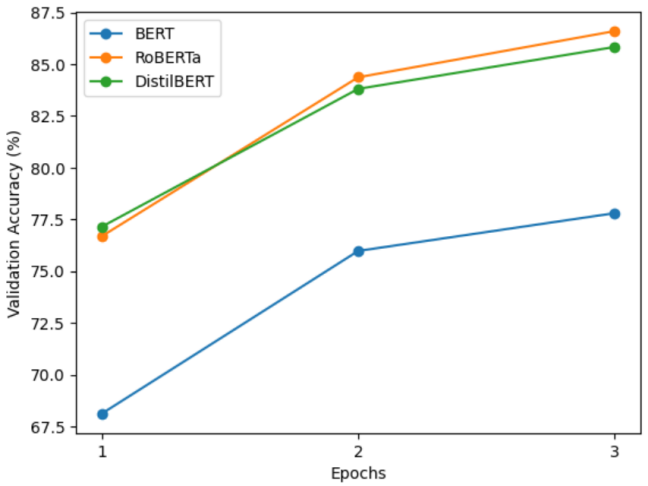
All Codes are in SC4001_distilBERTa.ipynb

epoch	Training Loss	Valid. Loss	Valid. Accur.	Training Time	Validation Time	F1 Score
1	0.117681	0.445852	0.856479	0:04:50	0:00:32	0.856452
2	0.182486	0.445852	0.856479	0:04:51	0:00:32	0.856452
3	0.222405	0.445852	0.856479	0:04:50	0:00:32	0.856452

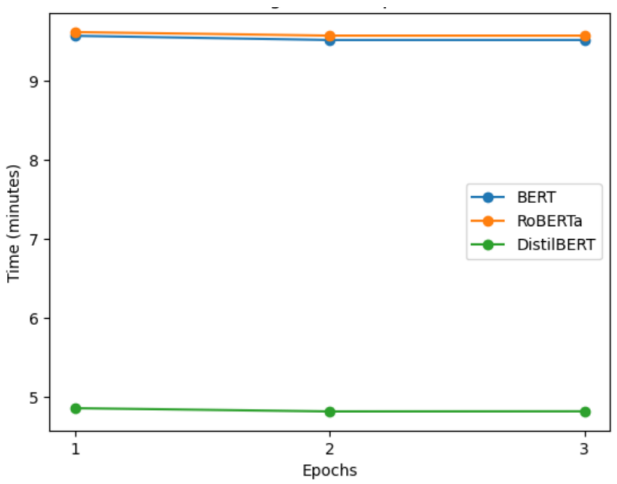
5. Result Comparisons

All Codes are in SC4001_ResultComparisons.ipynb

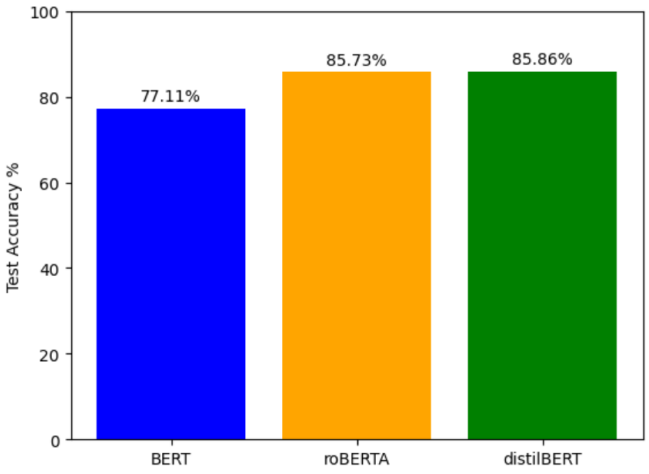
Validation Accuracy:



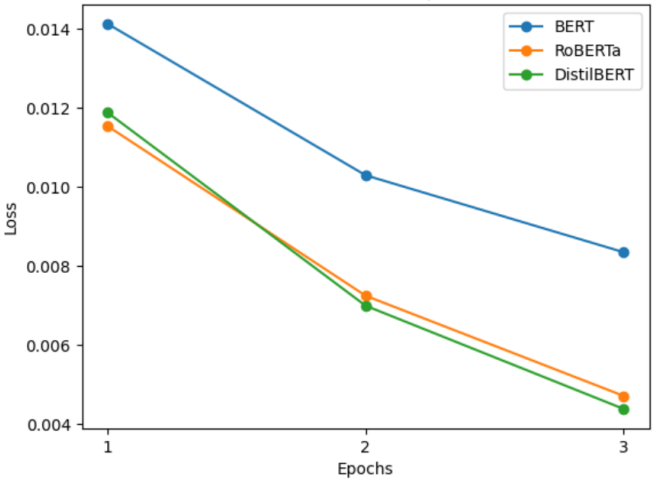
Training Time:



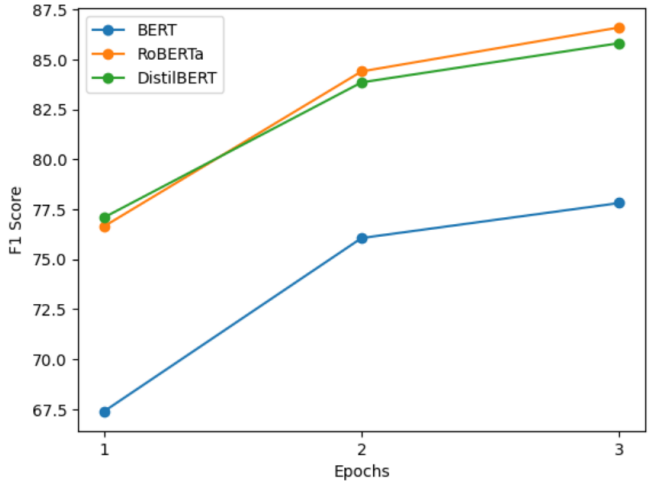
Test Accuracies:



Loss:



F1 score:



The F1 score is used to balance between precision and recall. It is useful in cases where there is uneven class distribution or when false positives and false negatives have different consequences. F1 score here is highest for distilbert and roberta.
For training time,distilbert is the fastest model.
For validation accuracy and test accuracy , roberta has the highest accuracy.
For training loss distilbert has the lowest training loss.

6. Conclusion

To sum up, RoBERTa performs the best but demands more time and computational resources. DistilBERT, while faster, remains effective for general sentiment predictions. The choice depends on the application: for a quick estimate, DistilBERT suffices, but for high accuracy, especially in critical tasks, RoBERTa is the preferred option. BERT falls in between, offering a balanced performance. CNN is not able to perform as well as these models due to the inability to capture complex relationships between words and to be fine-tuned on a variety of NLP tasks with limited data.

7. References

Rajapakse, T. (2019, October 13). *Simple transformers - multi-class text classification*. Medium. https://medium.com/swlh/simple-transformers-multi-class-text-classification-with-bert-roberta-xlnet-xlm-and-8b585000ce3a

Rajapakse, T. (2020, May 2). *Multi-class classification*. Simple Transformers. https://simpletransformers.ai/docs/multi-class-classification/

Wisdomml. (2022, August 5). *What are stop words in NLP and why we should remove them?*. Wisdom ML. https://wisdomml.in/what-are-stopwords-in-nlp-and-why-we-should-remove-them/#:~:text=They%20provide%20no%20meaningful%20information,data%20in%20terms%20of%20size.

Horev, R.(2018, November 11). BERT Explained: State of the art language model for NLP. Towards Data Science. https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270

https://www.analyticsvidhya.com/blog/2022/10/a-gentle-introduction-to-roberta/

Sharma, D. (2022, November 11). *Introduction to Distilbert in student model*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2022/11/introduction-to-distilbert-in-student-model/

Sharma, D. (2022a, November 9). *A gentle introduction to Roberta*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2022/10/a-gentle-introduction-to-roberta/

8. Dataset

https://www.kaggle.com/datasets/jp797498e/twitter-entity-sentiment-analysis