Task 1 report:

<u>Pre-process the images :</u>

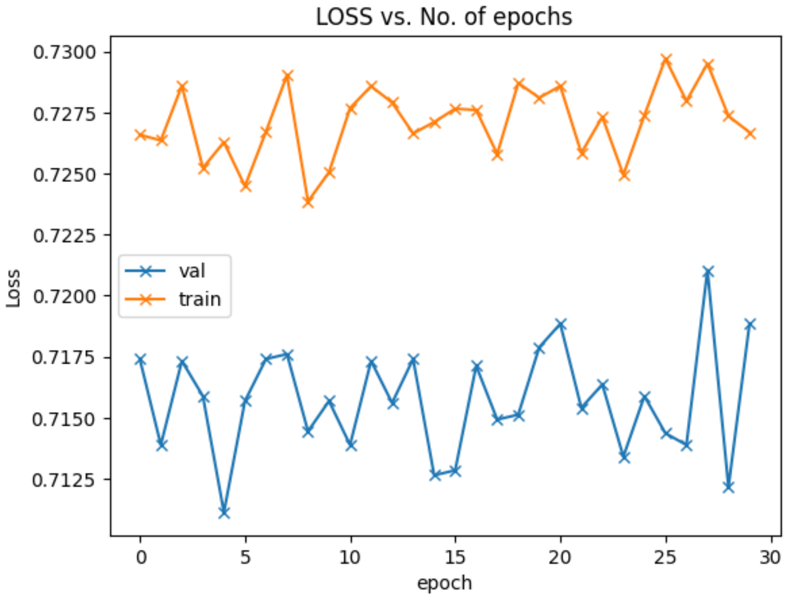
1) Resize to 128x128
2) Normalise the image: `mean=[0.485, 0.456, 0.406],std=[0.229, 0.224, 0.225]`

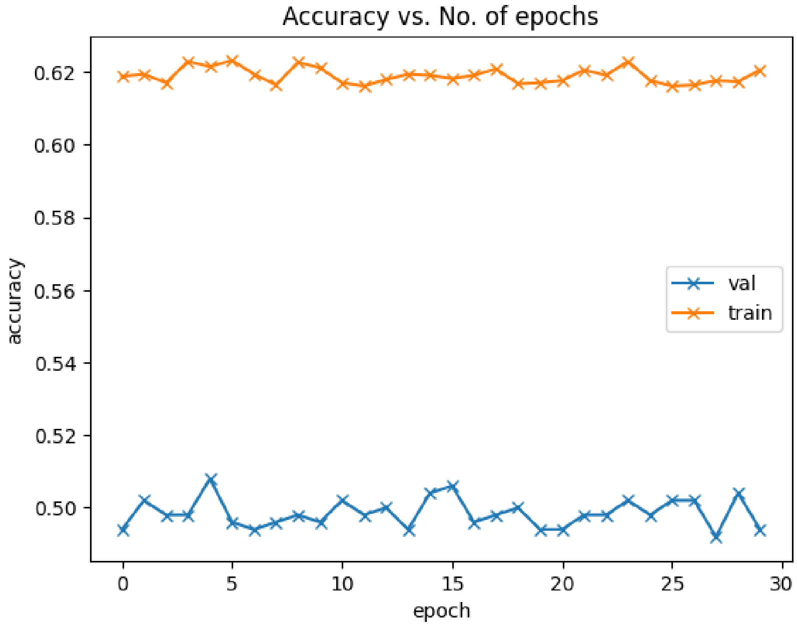
`Sample images after pre-process:`

0
20
40
60
80
100
120
0
25
50
75
100
125
don't think we forgot
i'm going to give you 30 seconds to delete this before i make the joke
back in the day
who did this
i win again!
Parkinson's disease
look at these assholes.
animals enjoy motorc
when you stuff your jewish chicken before you give it a good roast
those spooky white people
the leader of north korea during a meeting
i'm tired boss
you don't have to put on the red light
sea monkeys
i hate 3 things
racism sexism and white men
meanwhile in summer camp
when your dishwasher is done the good

<u>Model used</u> : Resnet50 with pretrain weights

Graphs:





The model's learning progress can be observed in the accuracy and loss graphs, but there appears to be minimal improvement in the model's performance. The accuracy graph shows a consistent range between 49 to 51%, and the loss graph follows a similar pattern. Despite the model's attempts to learn, there are no significant changes in its accuracy or loss values.

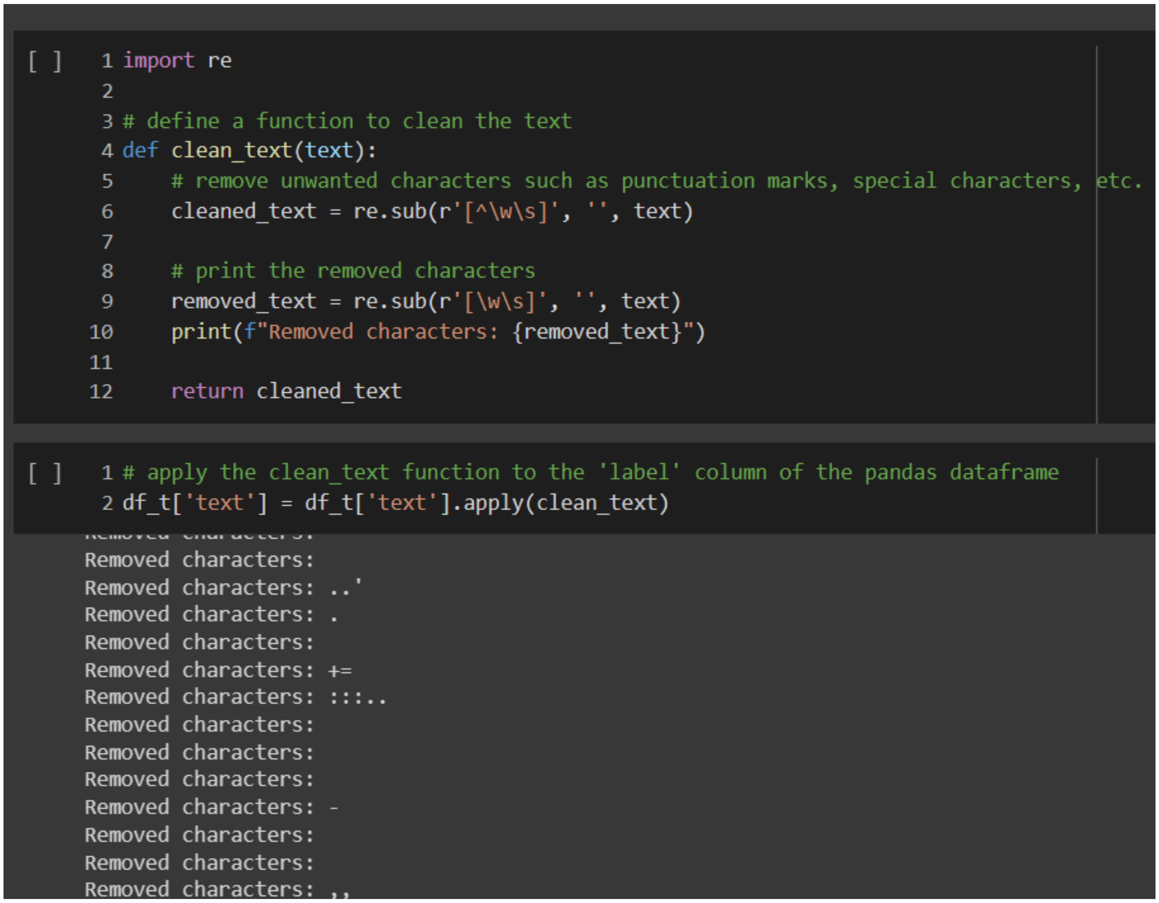
Results on evaluation step

Accuracy: 0.496
Precision: 0.455
Recall: 0.14693

F1 score: 0.222

Task 2 report:

Pre-process data :



```
import re

# define a function to clean the text
def clean_text(text):
    # remove unwanted characters such as punctuation marks, special characters, etc.
    cleaned_text = re.sub(r'[^\w\s]', '', text)

    # print the removed characters
    removed_text = re.sub(r'[\w\s]', '', text)
    print(f"Removed characters: {removed_text}")

    return cleaned_text
```

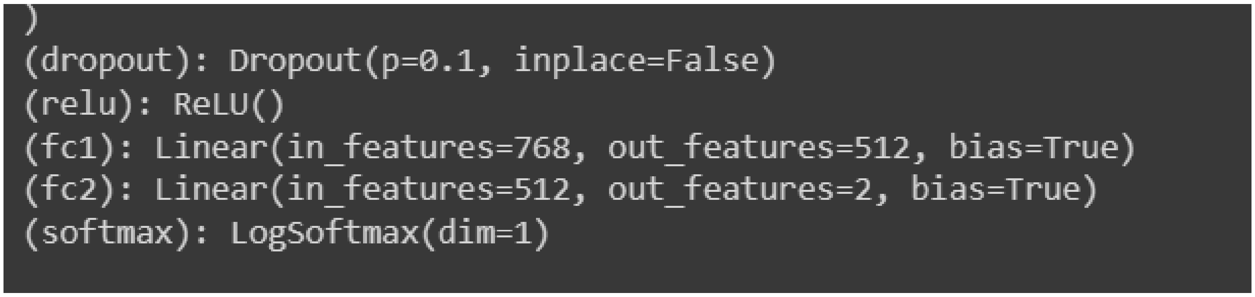
```
# apply the clean_text function to the 'label' column of the pandas dataframe
df_t['text'] = df_t['text'].apply(clean_text)
```

```
Removed characters:
Removed characters:
Removed characters: ..'
Removed characters: .
Removed characters:
Removed characters: +=
Removed characters: :::..
Removed characters:
Removed characters:
Removed characters:
Removed characters: -
Removed characters:
Removed characters:
Removed characters: ,,
```

Model used :

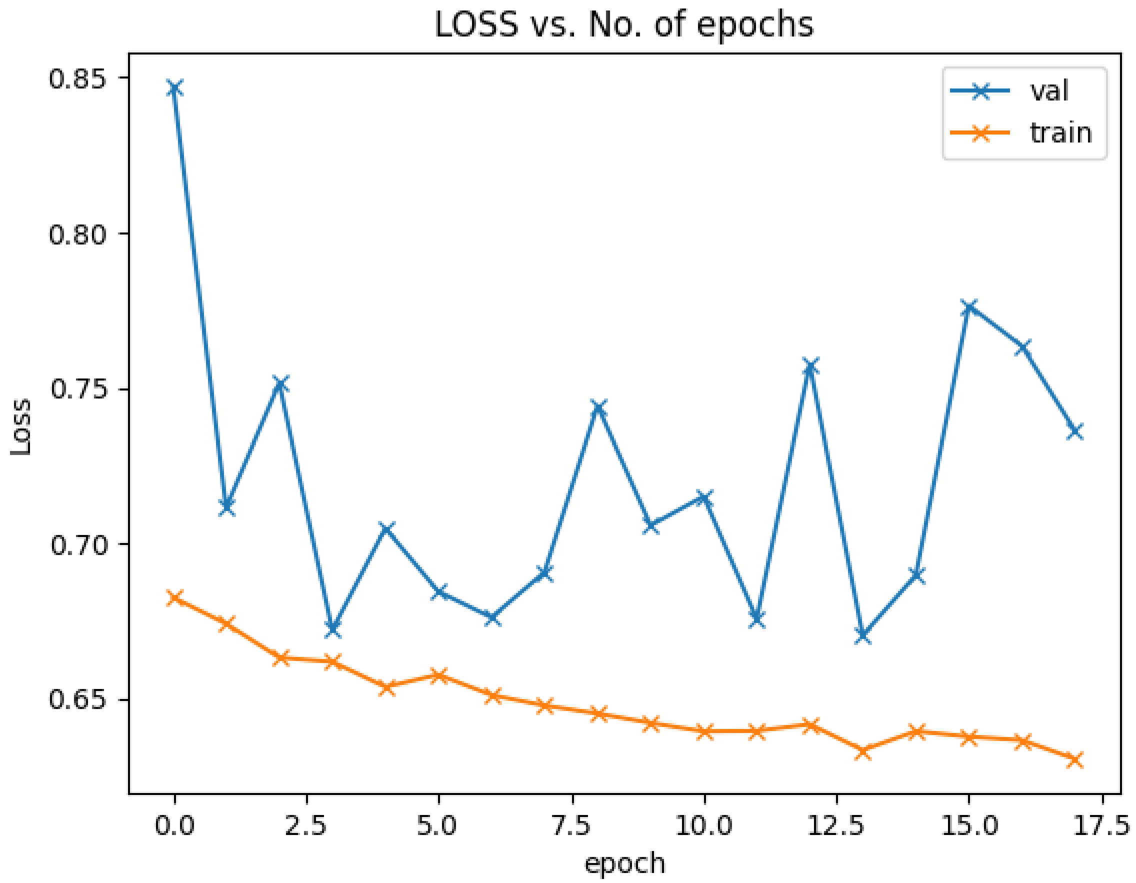
Pretrained BERT model has been used in this case. To train the model for the above Hateful and Not-Hateful memes, last-layer fine tuning has been done.

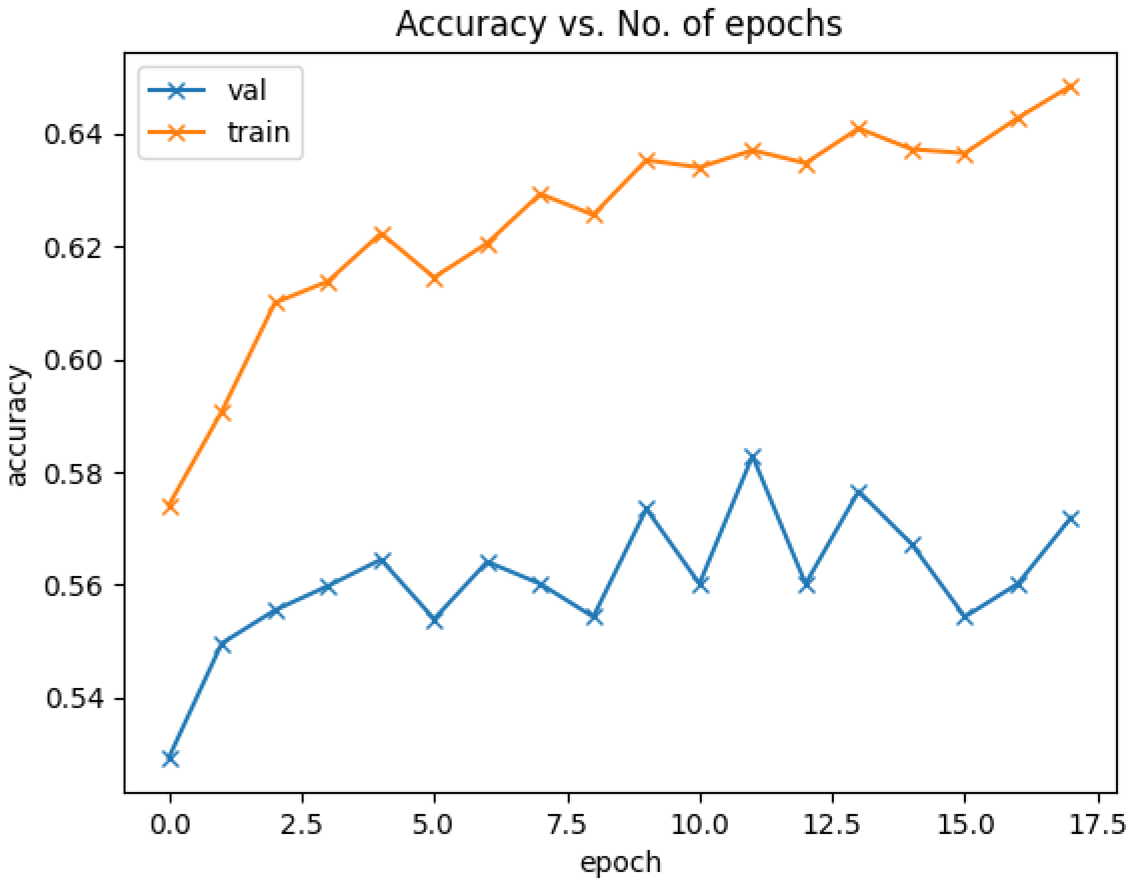
Last Layer:



```
)
(dropout): Dropout(p=0.1, inplace=False)
(relu): ReLU()
(fc1): Linear(in_features=768, out_features=512, bias=True)
(fc2): Linear(in_features=512, out_features=2, bias=True)
(softmax): LogSoftmax(dim=1)
```

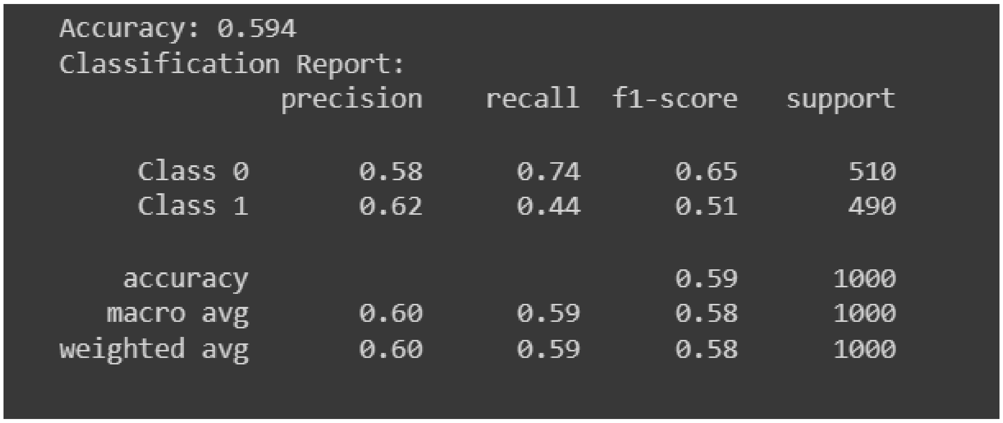
Graphs:





Result:

Accuracy: 0.594
Precision: 0.621
Recall: 0.43877
F1 score: 0.5143

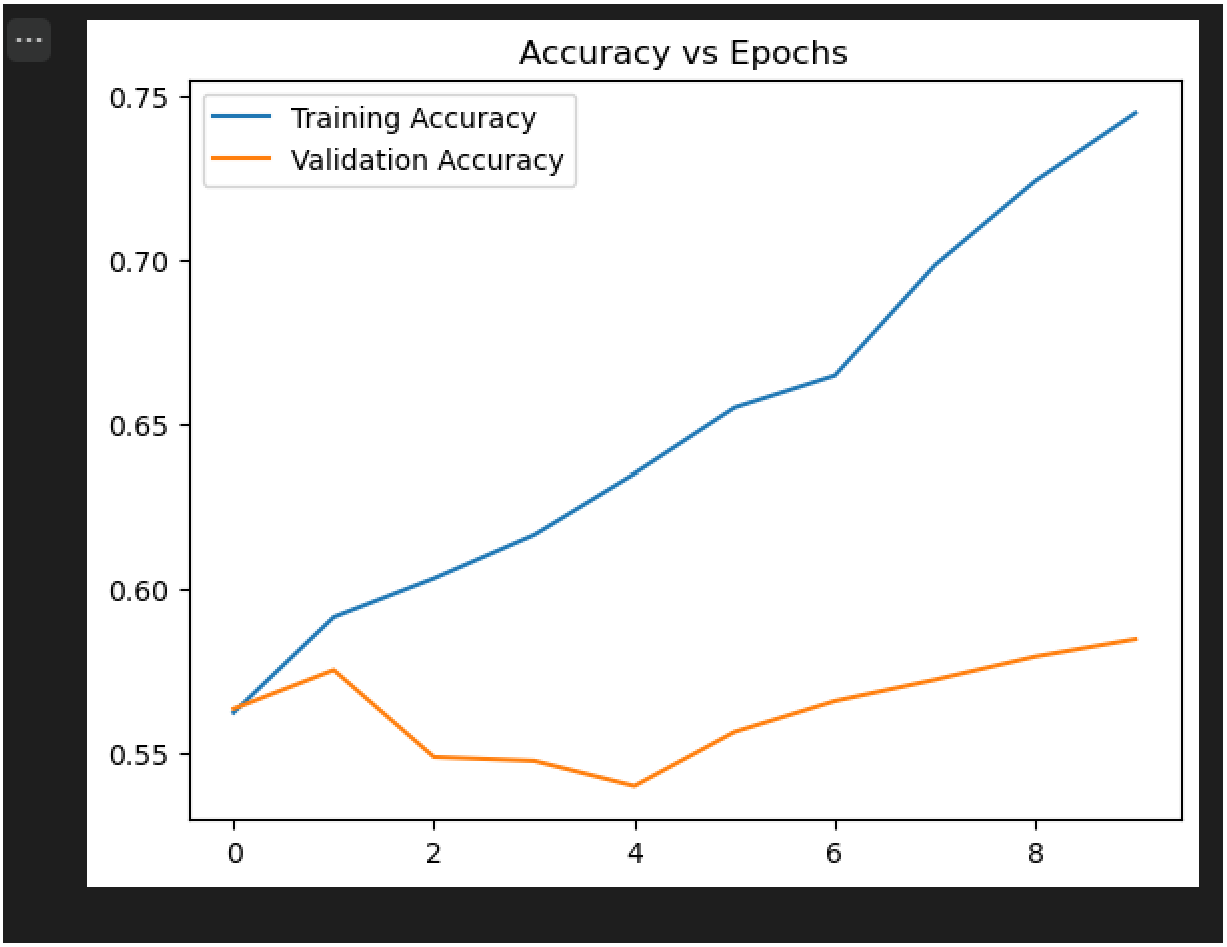


```
Accuracy: 0.594
Classification Report:
              precision    recall  f1-score   support

     Class 0       0.58      0.74      0.65       510
     Class 1       0.62      0.44      0.51       490

    accuracy                           0.59      1000
   macro avg       0.60      0.59      0.58      1000
weighted avg       0.60      0.59      0.58      1000
```

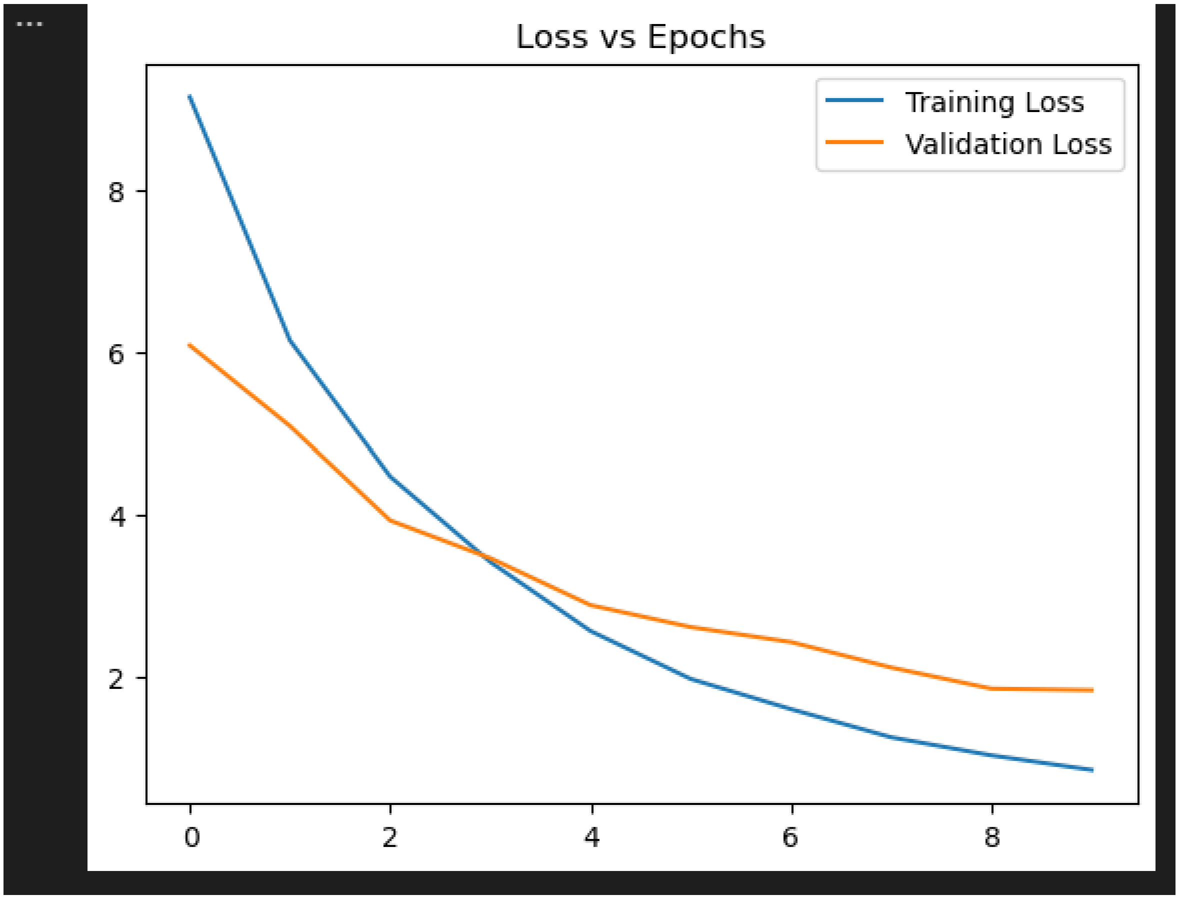
Task 3 report:

Graphs Of Loss and Accuracy

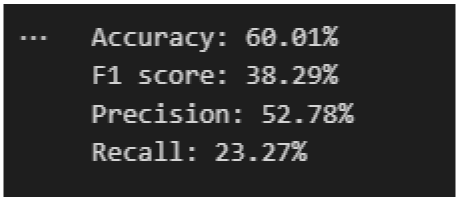
1) Accuracy vs Epochs



2) Loss vs Epochs

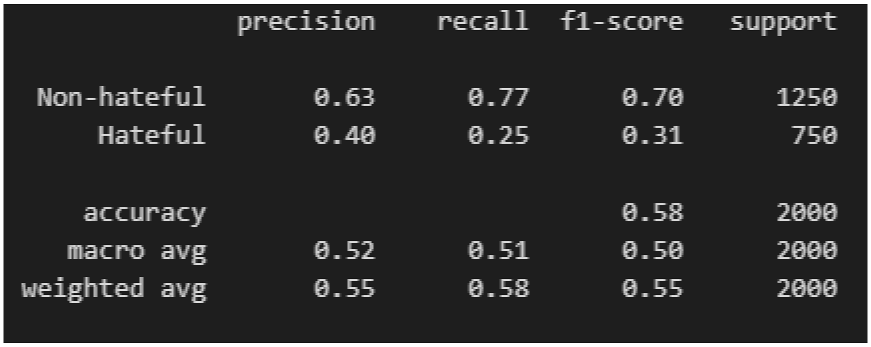


3) Scores:



```
Accuracy: 60.01%
F1 score: 38.29%
Precision: 52.78%
Recall: 23.27%
```

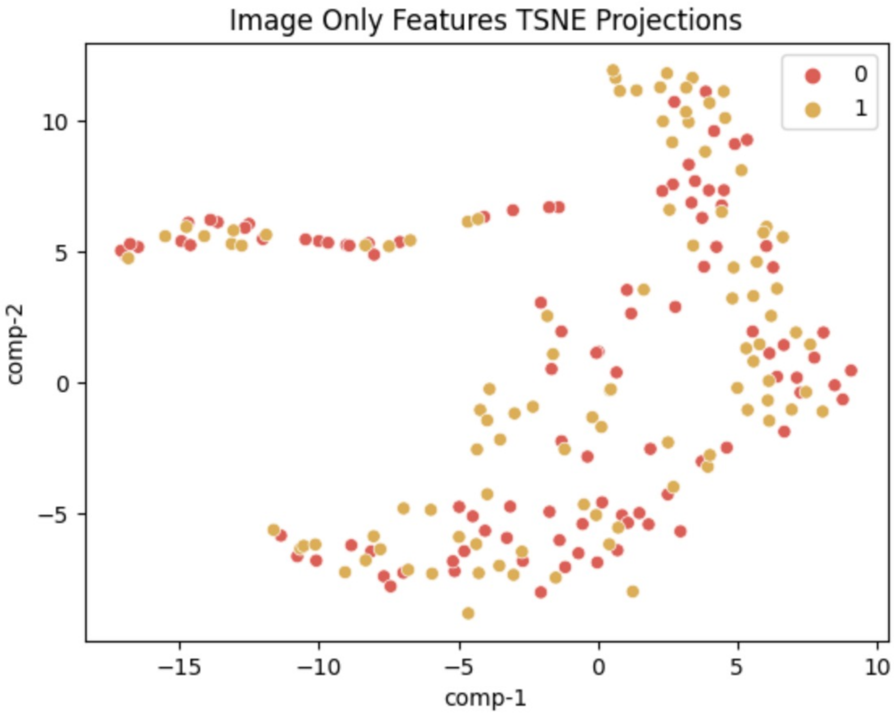
ClassWise Scores:



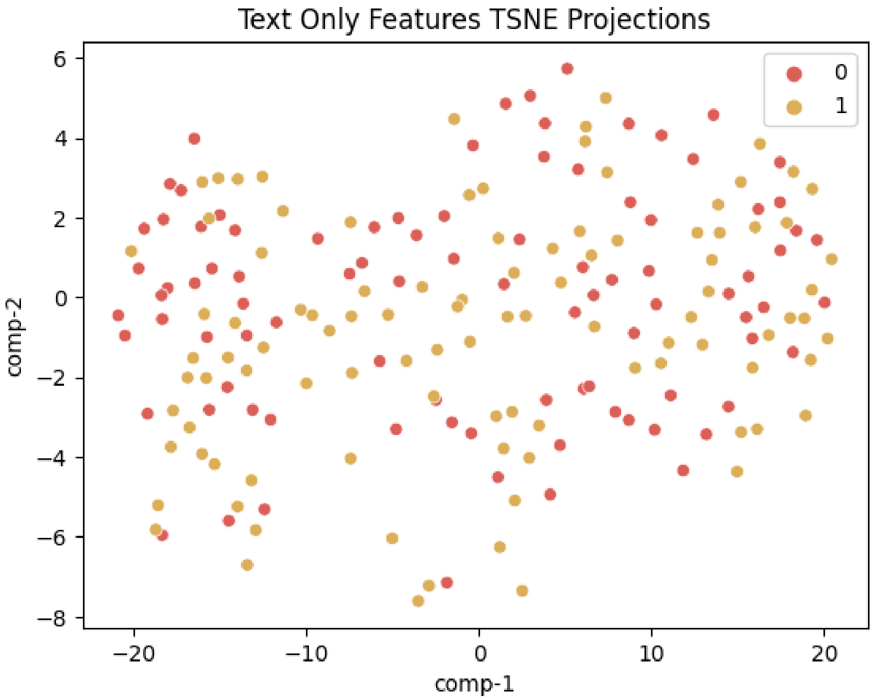
```
              precision    recall  f1-score   support

 Non-hateful       0.63      0.77      0.70      1250
     Hateful       0.40      0.25      0.31       750

    accuracy                           0.58      2000
   macro avg       0.52      0.51      0.50      2000
weighted avg       0.55      0.58      0.55      2000
```

4) TSNE PLOTS

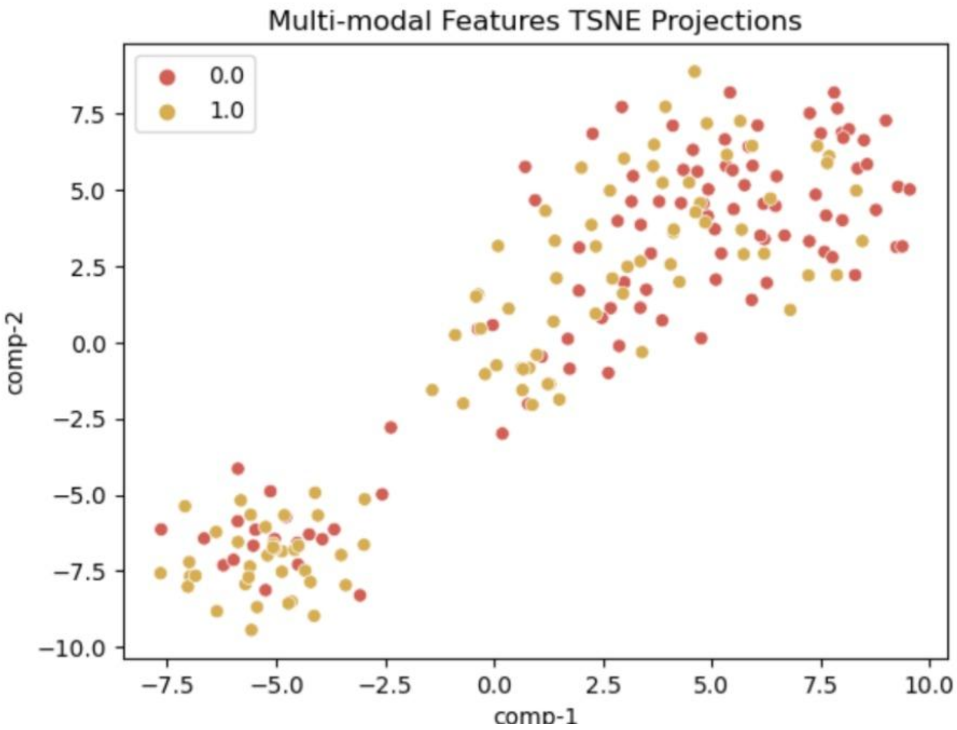
TSNE for task 1:



TSNE for Task 2:



TSNE for Task 3:



1. From the TSNE plot of the TASK 1 model, it is evident that the model is unable to determine the hateful and non-hateful memes separately. Hence, more features are required for better classification.

2. From the TSNE plot of the TASK 2 model, it is evident that the model is unable to determine the hateful and non-hateful memes separately, given the fact that the model is scattering the data evenly. Hence, more features are required for better classification.

3. From the TSNE plot of the TASK 3 model, although not clearly, but two clusters are being formed from the use of the model. Hence the multimodal is performing better than the previous two models evidently.