Approach to the Problem:

1. Preprocessing & EDA:

a) **Data Information Gathering:** The data primarily have no null or wrong values. So, null value removal and similar processes were not needed.

b) **Data Exploration via word count:** The training and validation data word count uniqueness has been tested using **WordCloud**. So important information was retrieved.

c) **Spell Check and high-level grammatical check:** The texts were also checked using TextBlob if any of them contains wrong spelling, but it was found that no error was present. When a sentence was passed in a similar way some error checks were evident by the **TextBlob** module, while checking the results and distribution with the target feature it was found that the distribution was uniform in both sets, thus indicating the technique might be misguided on the data.

d) **Use of transformers to generate feature attributes:** pre-trained **BERT** model was used to generate features from the existing dataset. In this section, we have carefully chosen tokenizer **BERT-BASE_CASED** as the case that might be delivering some insights into the prediction. The final dataset contained 768 features generated from the texts.

e) **PCA & t-SNE:** The next step contained a detailed feature Distribution check using dimensionality reduction techniques such as **PCA** and **t-SNE.** Surprisingly the distribution was spread similarly for both classes in a single cluster group. Hence, no information was retrieved.

f)

2. Training Data Preparation:

The training data and validation set were put into KFold for 10 and 5 folds respectively and taken for training. The training and validation data were both prepared to pass through model training.

3. Model Training & Evaluation:

Some of the most used models for classification for NLP were used - 1. RandomForestClassifier, 2. XGboost, 3. GaussianNB, 4. MultiNomialNB, 5. LightGBM.
From all of the models, the best results were found from MultinomialNB when trained on the validation chunks. But as it was only giving a biased output (for value 1), thus rejected. While the XGBoost model trained on a chunk of the train set was giving good results are can be called less biased than MultinomialNB.

The results came from the XGBoost model:

Macro Average	Micro average	F1 Score
0.54	**0.54**	**0.54**