Akhil\_Vallala\_Assignment\_3

# Introduction:

In sentiment analysis, we classify the polarity of a given text at the document, sentence, or feature level. It tells us whether the opinion is positive, negative, or neutral. If we go more advanced, like beyond polarity, we can go for emotional states like angry, sad, and happy.

In this comparison, two sentiment analysis models, TextBlob and VADER, are evaluated for their performance. Using pattern analysis, TextBlob—which is well-known for its simplicity and convenience of use—is especially useful for rapid prototyping and language translation projects. However, because of its sensitivity to both polarity (positive/negative) and intensity (strong) of emotions, VADER (Valence Aware Dictionary and Sentiment Reasoner) performs exceptionally well when managing sentiments expressed in social media situations. Because emoticons, punctuation, and capitalization are important in short-text online communications, VADER can now more properly read sentiment thanks to its incorporation of grammatical and syntactical norms.

# Data:

The dataset comprises 1,000 text entries classified as positive and 1,000 classified as negative, each having undergone prior sentiment analysis processing. Given that the data has been preprocessed, steps typically involved in initial data preparation—such as removing stop words, stemming, and lemmatization—are not required in this context. This allows for a direct focus on comparing the performance of sentiment analysis models like TextBlob and VADER on the curated dataset.

# TextBlob:

## count\_sentiment function:

Based on the TextBlob sentiment analysis, the `count\_sentiment` function determines how many text files in a given directory match a certain sentiment criterion. It returns the number of files with sentiment polarity above or below the threshold, denoting positive or negative sentiment, depending on the directory name, sentiment type, and polarity threshold entered as inputs.

### Code:

A screen shot of a computer program

Description automatically generated

## Recording values of number of identified reviews for all thresholds:

Using sentiment analysis, the code snippet divides text files into two directories into positive and negative categories. To find the accuracy of sentiment recognition with TextBlob, it cycles through polarity thresholds from -0.9 to -0.1 for negatives and 0.1 to 0.9 for positives. Assuming there are 1,000 files in each sentiment category, the results of successfully detected files for each threshold are recorded.

### Code:

A computer screen shot of a program

Description automatically generated

## Plot for a number of identified reviews vs thresholds:

A graph of positive and negative

Description automatically generated

## Observation:

By looking at the graphs, one can see that the accuracy of identifying negative sentiment is rather low across the studied polarity thresholds, and that it only becomes much higher as one approaches -0.1. It could be helpful to think about using a less negative (more positive) threshold in order to increase the sensitivity of negative sentiment identification. This change might make it possible to identify negative sentiment in the examined text files in a more balanced manner.

## Improvements:

Recorded the values from -0.3 to 0.2 for both positive and negative to find which threshold gives a maximum number of negative and positive identified files.

### Code:

A screen shot of a computer program

Description automatically generated

### Plot:

A graph with red and blue lines

Description automatically generated

### Observation:

The graph shows that the threshold value for sentiment classification is about 0.09 (just under 0.1). The polarity threshold represents the point at which the sensitivity of detecting positive and negative sentiment is optimal, as the two lines converge. The graph shows that setting polarity threshold around 0.09 for the current dataset would provide the most precise sentiment analysis results.

# Vader:

# count\_sentiment\_vader:

The count\_sentiment\_vader () function checks a list of text messages in a given directory to see how many contain a particular text type as defined by VADER's sentiment analysis. You can specify whether you want to see positive or negative text, and you can set a value for the threshold value to determine whether the text is positive or negative.

### Code:

A screenshot of a computer program

Description automatically generated

## Recording values of a number of identified reviews for all thresholds:

The code defines an analysis that measures the performance of VADER's sentiment analysis between threshold values ​​of -0.9 and 0.9. This justifies the two lists to store the negative and positive text detection results. For each threshold, count\_sentiment\_vader is called twice, once for files in the negative directory and once for files in the positive directory, and stores the number of files that meet the sentiment criteria in lists corresponding to each threshold.

### Code:

A computer screen shot of text

Description automatically generated

## Plot:

A graph with red and blue lines

Description automatically generated

## Observation:

The graph shows that the number of correctly identified negative files continuously decreases as the polarity threshold increases from -0.9 to 0.9. The detection accuracy of positive files remains relatively stable, and values slightly decrease as the threshold progresses. When the goal is to optimize the simultaneous detection of  positive and negative sentiment with high accuracy, the data show that a threshold value of -0.9 is most effective. This threshold gives the largest number of correctly identified files for negative emotions while maintaining strong performance for positive sentiment.

# Comparison:

The TextBlob model shows balanced performance in sentiment classification, with an optimal threshold of around 0.09, which slightly favors positive sentiment detection. This threshold is useful because it allows a small deviation from neutrality and ensures that the sentiment classification does not lean too much towards one polarity, maintaining a reasonable sensitivity to both positive and negative sentiment.

On the other hand, the VADER model requires a significantly negative threshold of - 0 .9 to achieve optimal performance in both positive and negative sentiment recognition. This extreme threshold indicates that the VADER is prone to detect negative emotions, perhaps at the expense of classifying positive emotions as negative.

# Conclusion:

Considering that sentiment analysis requires a nuanced approach where both positive and negative sentiments are detected with high accuracy, TextBlob is a more appropriate model for this dataset. TextBlob's near-neutral optimal threshold indicates a more balanced classification mechanism, while VADER's optimal performance with a strongly negative threshold indicates negative emotion detection. Therefore, TextBlob is best for this particular dataset.