📕

Text Classification

What is Text Classification?

Text classification is a fundamental task in natural language processing (NLP) that involves categorizing text into predefined categories based on its content. It is also known as text categorization or document classification.

Text classification's primary goal is automatically analyzing and understanding large amounts of text data. This can be useful in various applications such as sentiment analysis, spam filtering, topic modeling, and language identification.

Assignment Overview

In this assignment, I used Sklearn using text data to find the performance of three text classification algorithms. I will discuss my data and the performance of various approaches.

Implementation

Data Overview

FIFA World Cup 2022 Tweets

A Twitter dataset about the FIFA World Cup 2022

https://www.kaggle.com/datasets/tirendazacademy/fifa-world-cup-2022-tweets

I decided to use a World Cup tweets data set. I picked it because it showed the tweet in its text form and the sentiment of each tweet. I had to remove emojis and links to make the data easier.

Naive Bayes

The first algorithm I tried was Naive Bayes. It calculates the probability of a data point belonging to a particular class, given some observed features. The algorithm is called Naive because it assumes that the features are independent from each other.

Results

Naive Bayes was the algorithm that took the least time to train. It produced a score of 87.8%. I believe the score was good due to the fact that there isn't much data in the set to begin with. It might not have as big of an advantage in a larger dataset.

Logistic Regression

The following statistical algorithm I tried was Logistical Regression. This algorithm works by modeling the relationship between input and output variables, estimating the probability of using a logistic function.

Results

Logistic regression produced a slightly better accuracy score of 88.7%. However, it took longer than Naive Bayes, which makes sense because it requires more computationally heavy operations. A possible explanation of why it might've performed better is the nature of being able to model relationships between the input and output variables.

Neural Network

Neural Networks was the final algorithm I experimented with it was Neural Networks. They are inner workings inspired by the human brain. They organize tasks by adjusting connections between processing units named neurons. During training, the networks change connections between neurons to more accurately predict the output for new input data.

Results

Surprisingly, on the initial testing for Neural Networks, I obtained a score of 54.2%, which was less than I anticipated. I messed around with the parameters and got a way better score. My final results were:

```
accuracy score:  0.8924068366489213
precision score:  0.9046256556986171
recall score:  0.9115809706871696
f1 score:  0.9080899952130206
```

I believe it got a better score than both of the previous algorithms due to learning the complex relationships, which in this case are tweets and the output. Also, the ability to fine-tune the equation helped the results a lot. While NN produced the best results, it is also important to remember it took the longest, which may be an issue on large data sets.

Conclusion

The choice between Logistic Regression, Neural Networks, and Naive Bayes depends on the given data set and the problem we are attempting to solve. If the time and budget allow, try each algorithm and compare the results to pick the best one for each situation.