Comp550 Assignment 1 Report

Ye Yuan

Abstract

In this study, I compared the performance of four text feature extraction methods, namely n-grams, stemming, stop word removal, and lemmatization, in predicting whether a given sentence is in positive or negative sentiment. Stemming works a little better than unigram processing. Stopping words removal and lemmatization were done at the same level as unigram processing. On the test, their accuracies are very comparable. Furthermore, while bigrams and trigrams have lesser accuracy on the test set, they fit better in the training set.

Introduction

Sentences categorized as positive or negative are included in the dataset, and the overall purpose of the project is to make predictions on particular labels using the aforementioned data and various text feature extraction methods. We set the mean squared error (abbreviated as MSE) as the evaluation metric, as well as the accuracy score, and used logistic regression to analyze the datasets to achieve this purpose. Our most important findings are that stemming work better than others, and the trigrams perform worse.

Big Ideas

The following procedures were used to conduct the entire experiment:

1. Take care of the data sets

For the appropriate dataset, add a new column with 1 as the positive label and a new column with 0 as the negative label. Then, to conduct a random experiment, concatenate them as a single dataset and randomly shuffle them.

2. Divide your data into training and test sets.

Use sklearn's built-in function to randomly split the entire dataset into a training set and a test set, with the test set accounting for 15% of the total.

3. Use a variety of text feature extraction approaches.

Implement many types of text processing methods and convert them to vectors using the count vectorizer from sklearn.

4. Experiments with cross-validation

After processing the texts, the average MSE and accuracy score on both the training and validation sets were calculated using five cross-validations. Finally, the model was trained with the entire training set to predict the test set, and the MSE and accuracy scores on the test set were obtained.

Results

The following table displays the results. In general, stemming performs better on average validation accuracy, thus if we utilize it as the preprocessing method, we should expect greater performance on predicting unseen data. In this scenario, trigrams, on the other hand, produce the worst results.

Methods	Average Training Accuracy	Average Validation Accuracy	Test Accuracy	Average Training MSE	Average Validation MSE	Test MSE
Unigram	0.978288	0.758113	0.779375	0.021712	0.241887	0.220625
Unigram + Stemming	0.963088	0.759105	0.78125	0.036912	0.240895	0.21875
Unigram + Lemmatization	0.973626	0.757341	0.77375	0.026374	0.242659	0.22625
Unigram + Stop words removing	0.977709	0.752925	0.75875	0.022291	0.247075	0.24125
Bigrams	0.999531	0.702384	0.6975	0.000469	0.297616	0.3025
Trigrams	0.995752	0.607484	0.616875	0.004248	0.392516	0.383125

In comparison to simply employing unigram, stemming improves the performance of our model slightly, resulting in better average validation accuracies. The performance of a combined unigram with stop words removed, on the other hand, is slightly poorer, indicating that we cannot simply eliminate the stop words in this project. Even if stop words aren't effective for other document classification tasks, they do play a part in this one. Moreover, the results of the combined unigram with lemmatization and the pure unigram are comparable. Finally, we found that employing bigrams and trigrams resulted in decreased training mistakes but larger validation errors. This is most likely due to overfitting in the training set, as we are unable to perform effectively on the unknown data set.

Conclusion and Discussion

As expected, the choice of preprocessing methods has a direct relationship with performance. In this scenario, Stemming delivered the best results. Stop words removal and lemmatization produce nearly identical results. It's difficult to say which technique would always perform better in terms of accuracy on the test set. When compared to pure unigram, those two techniques have slightly lower performance, but all of the results obtained are satisfactory. Such inconsistencies can be ignored. For this classification challenge, bigrams and trigrams result in poorer accuracies. They would, however, be useful for other questions we might look at in the future. As a result, the only conclusion we can draw is that using stemming as a preprocessing strategy increases performance for this sentence sentiment classification problem.