DTSA 5509 Final Project

Yelp Reviews Classification

The problem

Yelp Reviews are an important part of a business. However, there can be a great amount of “fake” reviews that can be misleading to new potential customers and overall harm the reputation of the business. Therefore, the goal of this project is to develop a model that can help businesses be able to identify fake reviews that are harming their business and/or reputation. This is a binary classification problem that will be able to classify reviews as either “fake” or “real.

ML Approach

Data/Cleaning

The Reviews Yelp file contains information about reviews of businesses on Yelp, including the review ID, user ID, business ID, star rating, date, and text of the review. The file has 100000 rows (i.e., reviews) and 9 columns (i.e., features). The file size is 5341868833 bytes. The text of the review is a long string, while the other features are either categorical or numeric.

Modeling

- **Text preprocessing and Feature engineering on Reviews**
- **Train SVC Linear Model and Test it using our testing set**
- **evaluate out SVM classifier and see how it did in terms of accuracy.**

Results

Linear SVC: Accuracy: 0.48473967684021546

SVM Classifier : Accuracy: 0.8177737881508079

Final Results

```
Accuracy: 0.8177737881508079
```

```
F1-score: 0.9994514536478333
```

Conclusion

From the results above, using SVM modeling the model greatly improved from a Linear SVC model with an accuracy of 0.48 to an SVM classifier model with an accuracy of 0.82. Using grid search to find the best hyperparameters enabled the model to perform better in terms of accuracy compared to the previous model created. As for the F1 score we get a score of about 0.999.

For this project I explored a binary classifying Yelp Reviews as 'real' or 'fake'. First I did some data cleaning and exploration and then some preprocessing. I experimented with different algorithms, including linear svc, gradient boosting, SVM, and grid search cross validation to tune the hyperparameters for the final SVM classifier. The results show that I was able to achieve a relatively high accuracy of 82% on a balanced data set with about 1,000 reviews per star. However, there could be some further improvements such as experimenting with different text preprocessing techniques. There were also some limitations such as computer memory issues and not being able to utilize the full dataset in the millions. I think being able to use the full dataset would've resulted in better scores but even with the reduced data I was still able to reach a reasonable accuracy.