Supervised Learning To Detect Fake Reviews

CS 175 Winter 2020

List of Team Members:
Andrew Self, 44516972, awself@uci.edu
Brandon Teran, 39801033, bteran@uci.edu
Salvador Villalon, 24141795, salvav1@uci.edu

1. Project Summary

Online reviews often influence consumers' decision to buy or not to buy a product. However, according to a study from Stony Brook University, as many as "20% of Yelp's reviews are fake" (Rayana). Using various machine learning methods to detect fake reviews could allow websites to significantly reduce this number. In this project, we combine various word representations, classifiers, and architectures to 3 different data sets to determine the most effective method of identifying fake reviews.

2. Datasets

Yelp Review Dataset
Research Paper Used:
http://shebuti.com/wp-content/uploads/2016/06/15-kdd-collectiveopinionspam.pdf

This dataset includes a collection of Yelp reviews for hotels and restaurants in Chicago and New York. It consists of three real-word review datasets from Yelp.com with filtered (spam) and recommended (nonspam) reviews. The Chicago hotel reviews have 5,854 rows. The Chicago restaurant reviews have 61,541 rows. The New York hotel and restaurant review combined are 359,052. The dataset is in the form:

Review Text	Date	Review ID	Reviewer ID	Product	Label ("d" or "t")

Dianping Chinese Reviews Dataset
Research Paper Used:
https://www.cs.uic.edu/~liub/publications/fake-PU-learning-paper274.pdf

The second dataset is a CSV file that contains 9765 rows of data. These were reviews taken from Dianping. Dianping is the largest Chinese review hosting site, in the research they present

the first reported work on fake review detection in Chinese with filtered reviews from Dianping's fake review detection system.

Review Text	Label ("d" or "t")

Opinion Spam (OP Spam) Dataset
Research Paper Used:
https://homes.cs.washington.edu/~yejin/Papers/acl11_deception.pdf [Research Paper 1]
https://www.cs.cornell.edu/home/cardie/papers/NAACL13-Negative.pdf [Research Paper 2]

This corpus consists of truthful and deceptive hotel reviews of 20 Chicago hotels. This corpus contains: 400 truthful positive reviews from TripAdvisor (described in [1]), 400 deceptive positive reviews from Mechanical Turk (described in [1]), 400 truthful negative reviews from Expedia, Hotels.com, Orbitz, Priceline, TripAdvisor and Yelp (described in [2]), 400 deceptive negative reviews from Mechanical Turk (described in [2])

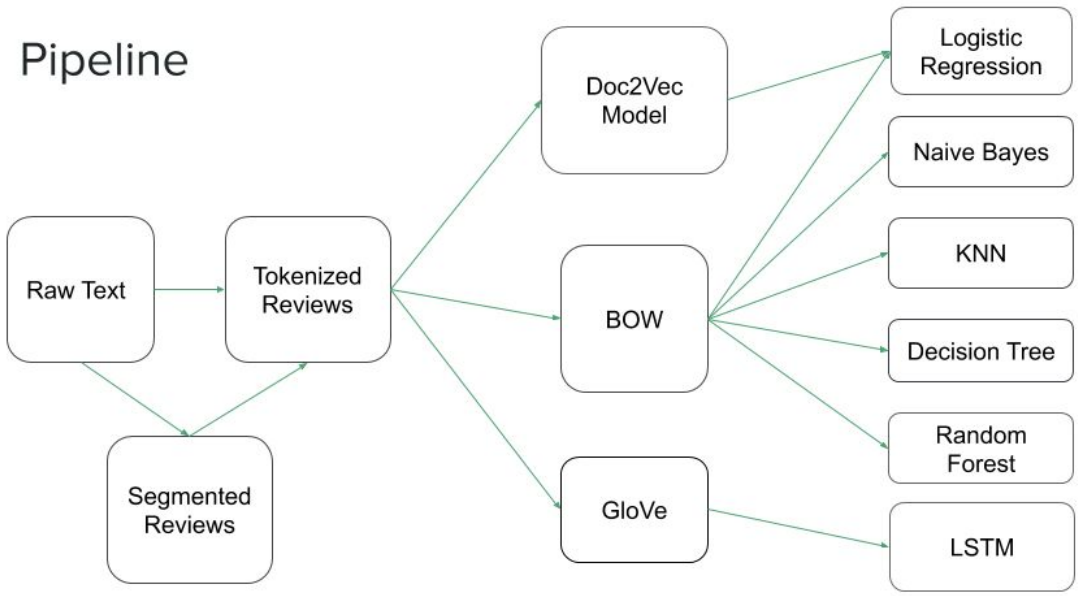
3. Technical Approach

Approach for each dataset

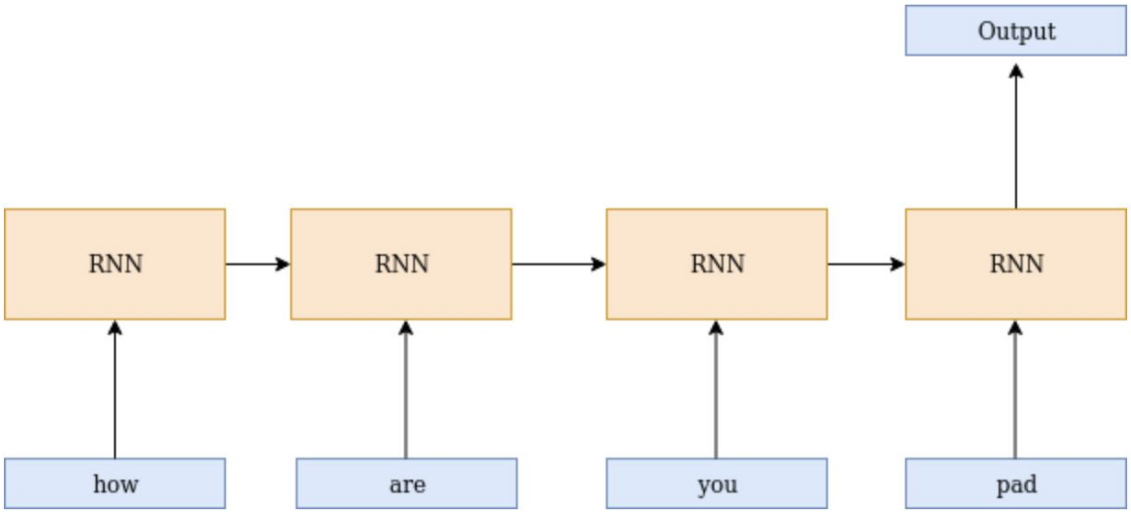
First we use CountVectorizer from the sklearn library to turn each review into a BOW representation. The CountVectorizer automatically segments English words, but for the Chinese dataset we used Stanford Word Segmenter. We then train classifiers using the BOW representation of each review and their label. We use the following classifiers from the sklearn library:

- Logistic Regression
- Naive Bayes
- K Nearest Neighbors
- Decision Tree
- Random Forest

In addition to the BOW representation, we also used Doc2Vec and GloVe embeddings. The Doc2Vec reviews were trained using logistic regression and the GloVe embeddings we trained using an LSTM Neural Network. Here is pipeline:



And here is the LSTM architecture:

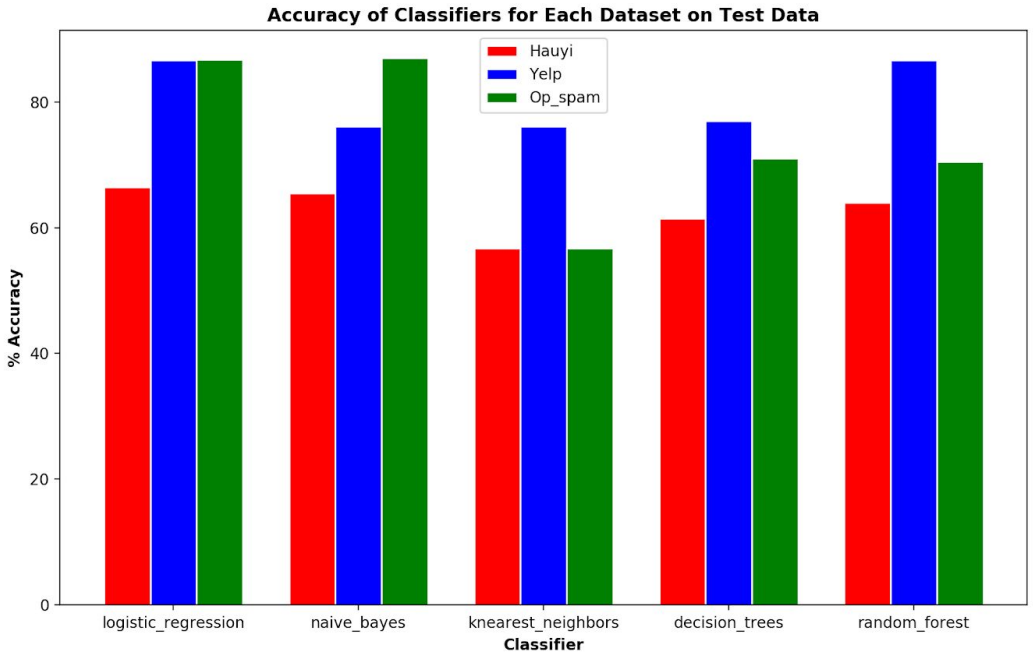


Inspiration for this architecture came from this tutorial:

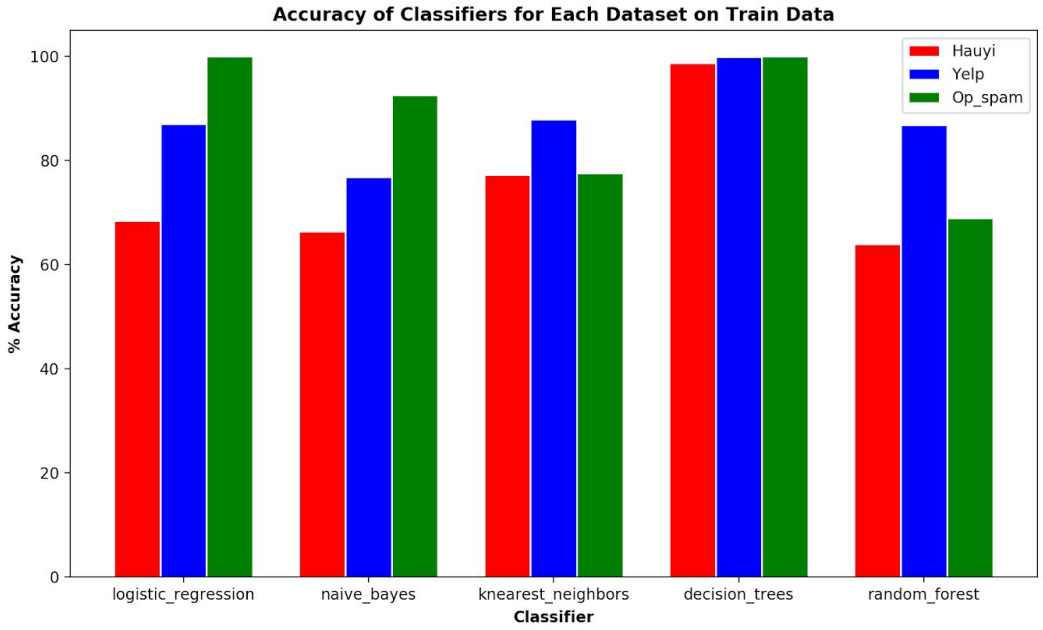
https://www.analyticsvidhya.com/blog/2020/01/first-text-classification-in-pytorch/

4. Results

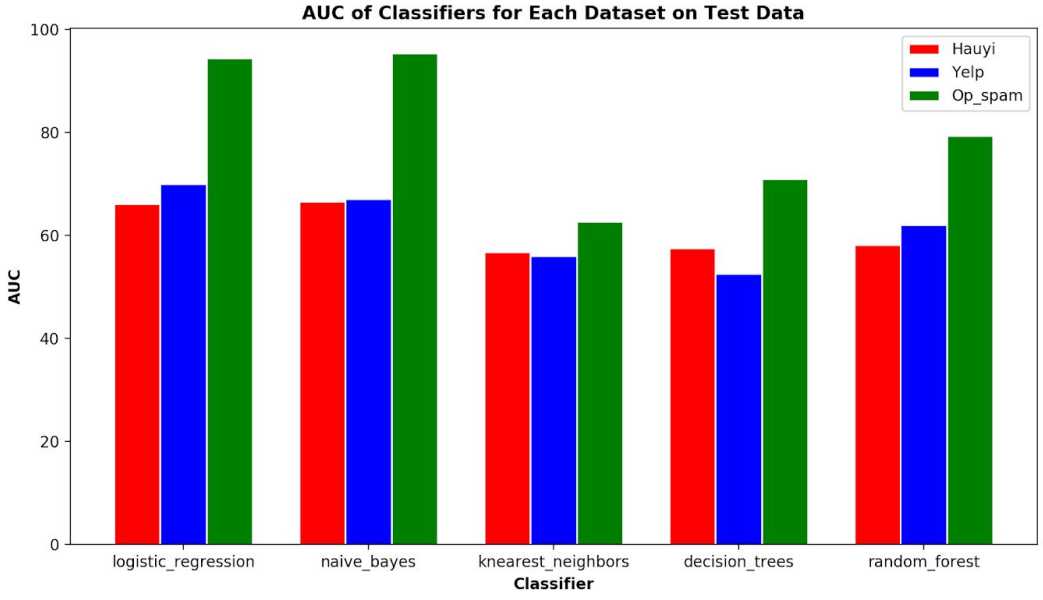
The following 4 charts show the effectiveness of each classifier on each data set. It is divided by testing versus training data and accuracy versus AUC as a metric. The following 4 charts also are all using the BOW representation.



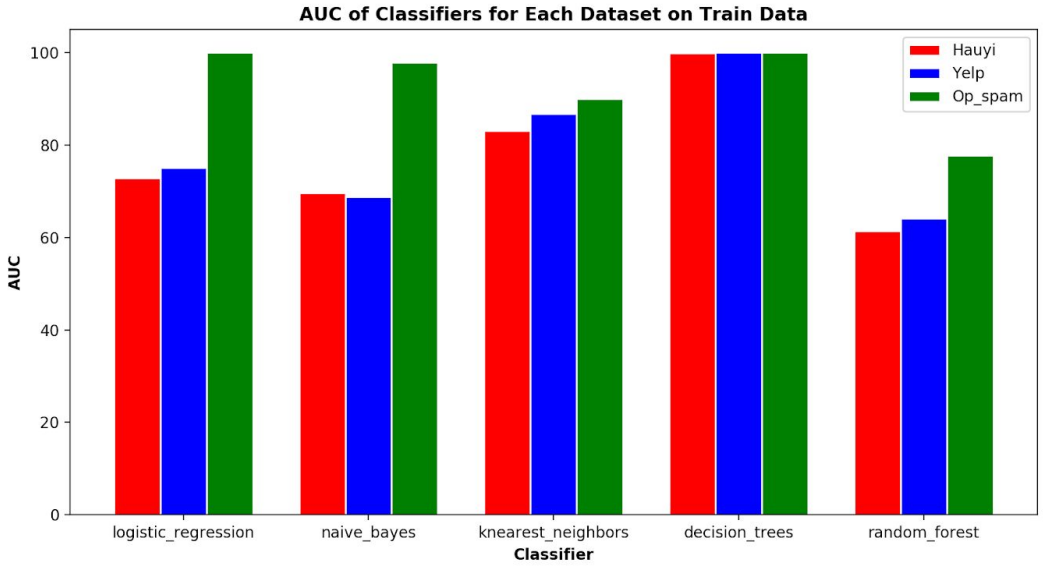
Here we can see the logistic regression classifier had the best average of all 3 data sets. Furthermore, the Yelp data set outperformed or tied the others on every classifier. I believe this is because it is larger and the algorithms are underfitting the other 2.



It is notable on this graph that decision trees are able to almost perfectly learn the training data but do not perform as well on the testing data. It is likely that the classifier is overfitting. Early stopping could be implemented to counteract this.

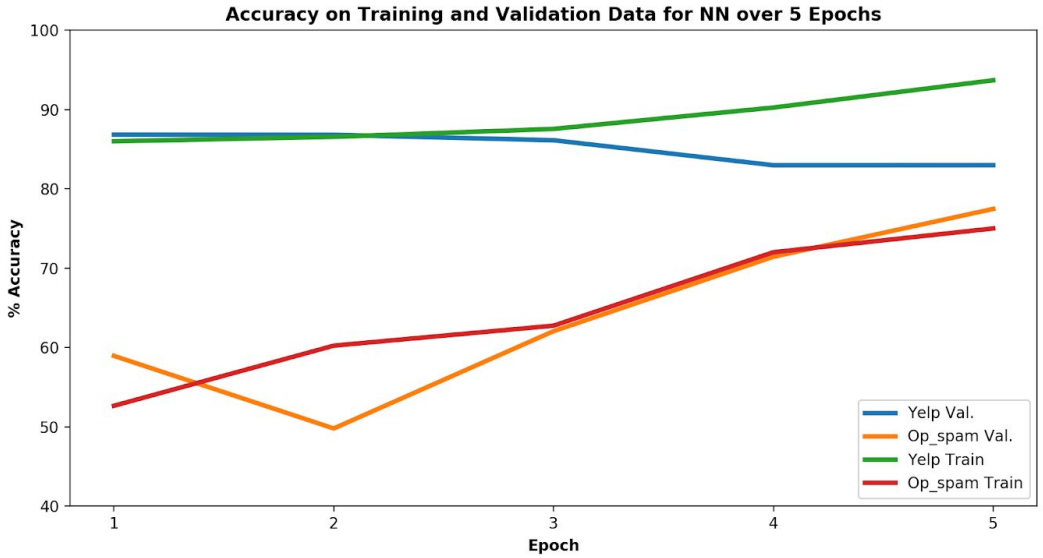


The Op_spam data set does the best on average when AUC is the metric and does especially well when using logistic regression and naive bayes.



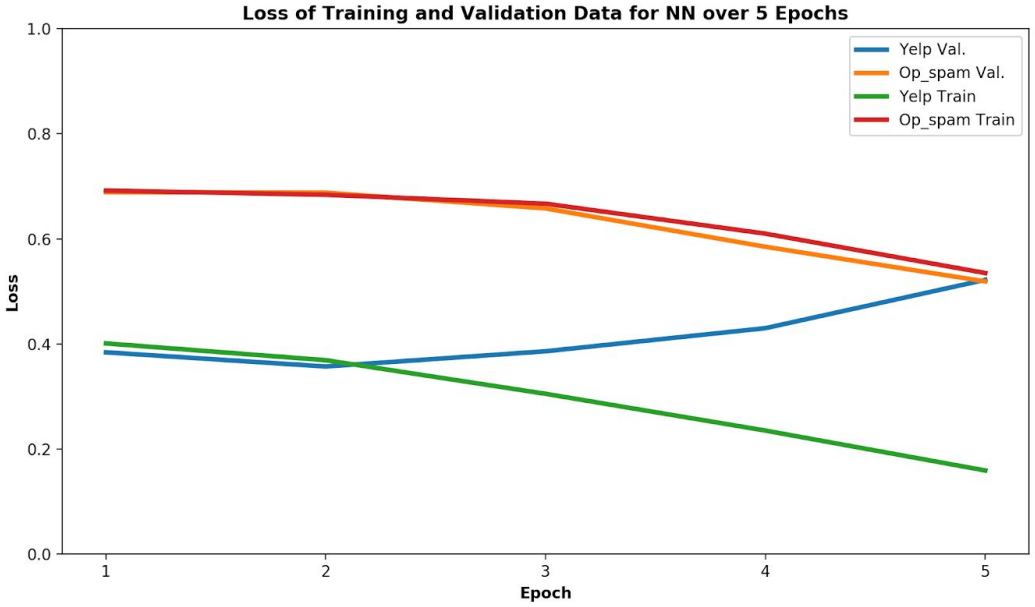
On the training data, we see a similar trend for AUC as we did with accuracy, where the decision tree classifier appears to overfit. Again, early stopping could be implemented to fix this.

This next chart shows data from the LSTM neural network we trained for this project. We did not use this method for the Chinese data because we were unsure how embeddings would work in a different language, but would like to explore that in the future. This chart shows the accuracy, on average, increasing each epoch of training.



The Op_spam accuracy starts lower and slowly catches up while the Yelp remains high.

This chart shows the loss, on average, slowly decreasing as each epoch progresses. We used Binary Cross Entropy (BCE) Loss and the Adam optimizer.

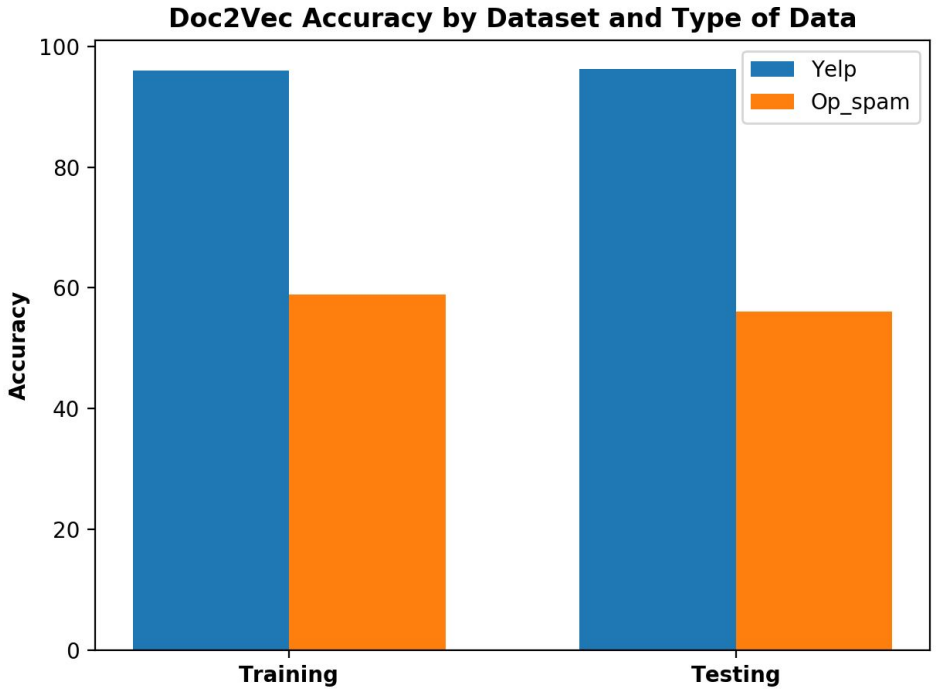


The yelp validation goes up. This could be fixed by trying a different optimizer or adding more epochs. Also, as to be expected based on the previous chart, the Op_spam data starts higher and then slowly moves down. Also, while not included in the chart, here are the results of the test data:

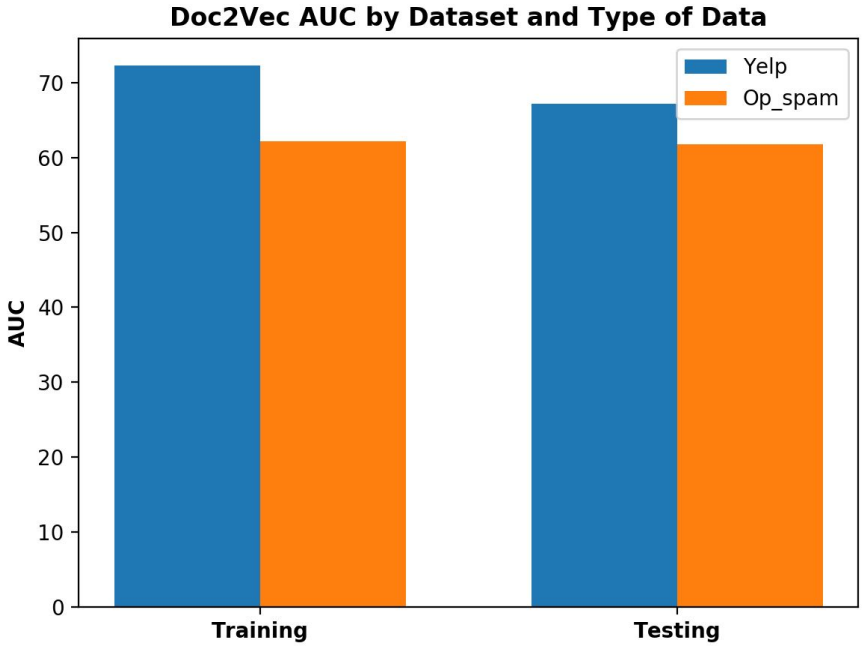
Op_spam Test Data
Accuracy: 70.76%
Loss: 0.552

Yelp Test Data
Accuracy: 82.84%
Loss: 0.519

This chart shows the accuracy for the Doc2Vec -> Logistic Regression for Yelp and Op_spam on testing and training data:



Here, we can see that Yelp performs very well and outperforms Op_spam significantly. This is likely because it is a larger dataset. This next chart is similar but with AUC instead of accuracy:



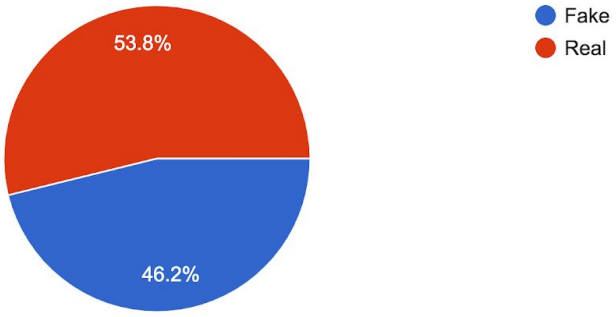
This time yelp only slightly outperforms Op_spam.

5. Survey

We asked 26 friends to label 4 reviews from the OP Spam dataset and got the following results:

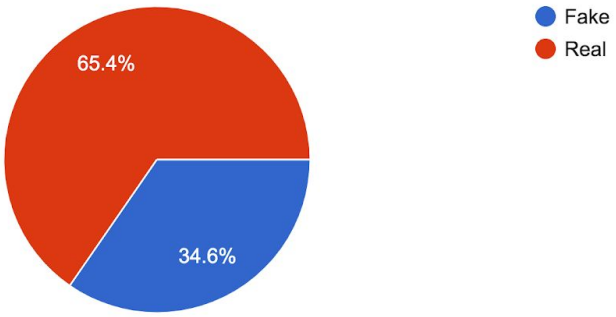
Review 1 (Truthful):

We stay at Hilton for 4 nights last march. It was a pleasant stay. We got a large room with 2 double beds and 2 bathrooms, The TV was Ok, a 27' CRT Flat...lways use the Michigan Av exit. Its a great view.
26 responses



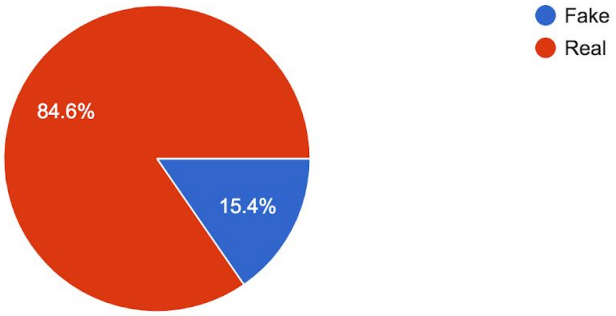
Review 2 (Deceptive):

My husband and I satayed for two nights at the Hilton Chicago,and enjoyed every minute of it! The bedrooms are immaculate,and the linnens are very... pool. I would recommend staying here to anyone.
26 responses



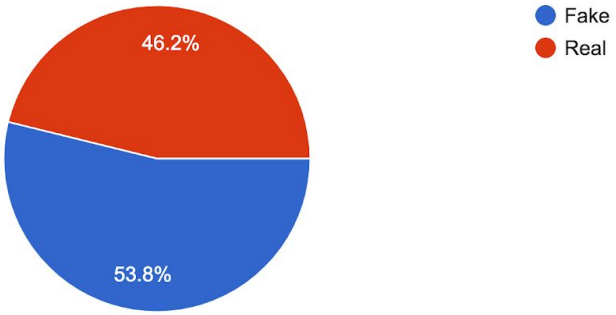
Review 3 (Truthful):

I recently stayed here for the Chicago triathlon. This was my third stay at this hotel. I have not had any issues until this visit. After the 3rd day of m...re many hotels to choose from in downtown Chicago.
26 responses



Review 4 (Deceptive):

The Hilton Chicago, located on prime real estate in downtown Chicago, has not aged as gracefully as other hotels around the country opened during ... paying a premium for a prosaic hotel experience.
26 responses



These results are consistent with the claim made in the aforementioned Stony Brook University paper “that humans are only slightly better than random (Rayana)” when it comes to detecting fake reviews.

6. Conclusions

After comparing the different data sets and methods, we have found a few key takeaways. First, the Yelp data set consistently outperformed the other two datasets, which were also smaller in size. This highlights the importance of having enough data to train a model to

perform complex tasks, such as text classification. Unfortunately, labeled spam detection data is scarce. Furthermore, our Doc2Vec to logistic regression model had the highest accuracy, followed shortly by BOW/ logistic regression and BOW/ random forest. Naive Bayes was also able to work very well on the smaller Op_spam dataset. Naive Bayes relies heavily on the appearance of certain words as predictors opposed to the count of words. This indicated that the appearance of certain words in the Op_spam data set word strong predictors of either truth or deception.

7. Appendices

1. Software

Publicly-Available Code
Programming Languages: ● **Python 3.7**, standard libraries, and public libraries including **NumPy, scikit-learn, Matplotlib, NLTK, Gensim, Torchtext, Matplotlib** **Preprocessing Software:** ● Tokenize the text with **NLTK** and exclude any whitespaces ● **LabelEncoder** from scikit-learn to normalize labels. Need to use this for Doc2Vec Model ● Chinese is standardly written without spaces between words. Used the **Stanford Word Segmenter** to split Chinese text into a sequence of words ● **Doc2Vec from Gensim** to help create numeric representation of a document **Evaluation Software:** ● Created a **Google Form** to allows others to help us see which reviews they think are fake