ORIE 4741 Midterm Report

Mina Huh (mh859), Matthew Danbury (mjd358)

November 1, 2021

1 Dataset Selection, Cleaning, and Exploratory Analysis

We initially planned on using the "Internet news data with user engagement" dataset but after some quick exploration found that there was a lot of missing data. Just by dropping the rows with missing values, roughly 72.8 percent of the data would be retained. In addition, some of the top news sources completely lacked the data for engagement reaction/shares/comments which further cut the available data down. As a result, we decided to go with a different data set: the UCI Online News Popularity Dataset (https://www.kaggle.com/thehapyone/uci-online-news-popularity-data-set) that has 2418284 records and 60 features present. While this was admittedly a little disappointing, it still gives us every opportunity to demonstrate to our intended client, a major news publication, the viability of incorporating these data tools into their business operations.

With this new dataset we started by using .describe() to get some general statistics, primarily looking out for missing values or values that don't make sense. Through doing so we found unreasonable minimum/maximum values for 'n tokens content', 'n unique tokens', 'kw avg min', 'kw min avg', and 'kw min min'. As 'n tokens content', 'kw avg min', 'kw min avg' and 'kw min min' all have to do with counts of tokens or shares, we filtered out the values that were negative. In addition, as 'n tokens content' is counting the number of words in the content of the article, we felt that a value of 0 did not make sense and found it to be an outlier to filter out as well. In the case of 'n unique tokens', we noticed a max value of 701 even though the field is measuring the rate of unique words in the content of the article. After filtering this max value out we checked the histogram and found that the remaining values correctly fell between 0 and 1 (Figure 1) When doing the filtering we also realized that the names of each column had an extra space in front so we used a replace function to modify each column name to not have extra spacing. From doing this, our dataset was reduced to 2304946 records, keeping around 95.3 percent of the initial dataset.

As we want to train a model to predict the popularity of an article, we will use 'shares' as the metric to determine popularity. After performing a 75-25

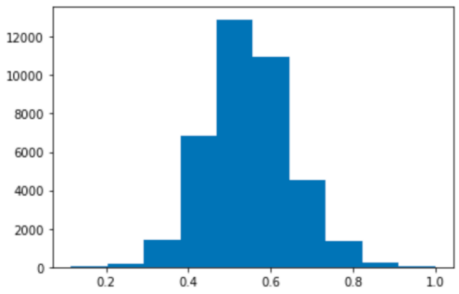

Figure 1:

train-test split, we made a histogram of the shares in the training set. The first thing we noticed was that on a linear scale, the distribution is very skewed to the right. However, after applying a log scale we can see that it more closely resembles a normal distribution (Figure 2). 1)

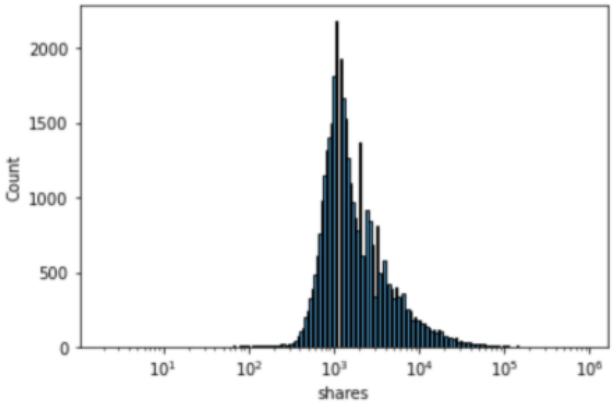

Figure 2:

2 Preliminary Results

We started with attempting a linear least squares regression model to predict the number of shares. This performed particularly poorly on the tails of the distribution (i.e. the articles with particularly high or low engagement) which is realistically what our news journalism publication client is most interested in. Because of this we decided to shift gears and turn this into a classification problem by dividing the training labels into quartiles, assigning the label "low engagement" if the number of shares of an article was below 1000, the label "average engagement" if the number of shares fell between 1000 and 3000, and

the "high engagement" if the number of shares exceeded 3000. We then applied the same mapping to our test labels. It is worth noting that none of the test examples or their summary statistics were used in deriving these boundaries so as to prevent data leakage in our model validation.

After transforming our data, we fit default scikit-learn implementations of ridge classification and random forest classification models to our training data and evaluated them on our test set with the (adjusted) balanced accuracy scores, and computed their confusion matrices. We chose these metrics so as to make sure our model didn't simply appear to do well by classifying every example as the dominant label "average engagement".

Ridge classification performed nominally the same as random guessing (technically 5 percent better) with confusion matrices [[[6998, 304],[2052, 262]], [[6886, 273],[2242, 215]], [[609, 4162],[445, 4400]]] for the three respective classes, whereas random forest classification performed marginally better than random guessing (14 percent) with confusion matrices [[[6819, 483],[1834, 480]], [[6555, 604], [1820, 637]], [[1333, 3438], [871, 3974]]] for the three respective classes.

3 Outlook

Getting performance of 14 percent better than random guessing out of the box (no hyperparameter optimization, additional feature engineering, etc.) was encouraging, especially considering the noisy nature of the problem at hand of predicting social engagement as a function of news content. We have begun to explore other ensemble methods such as boosting or ControlBurn. We also like tree methods for their interpretability as far as which features are the most significant as this would help our client understand what aspects differentiate high engagement from low engagement articles.

Over the course next month, we hope to optimize a tree ensemble model for classifying user engagement on this dataset. If time permits, we would like to attempt additional strategies for improving the interpretability of our model, such as model distillation, as well as exploring other user-facing or internal data tools for news publications.

In searching for datasets for this project, we came across a number of large datasets consisting of article headlines and publication times. One idea we had in terms of is to try to leverage unsupervised learning to understand the attention cycle of major news media. Specifically, we would like to perform transfer learning by feeding these headlines into a pre-trained neural network such as the Universal Sentence Encode. After performing this Euclidean embedding, we would attempt clustering on these embeddings and their publication times to extract patterns in media coverage.