LING 413 Project 2: Unsupervised Model On Sentiment Analysis with K-Means

Hao Qi (haoqi3@illinois.edu)

Abstract

Sentiment analysis plays a pivotal role in understanding consumer opinions and preferences, particularly in the realm of online reviews. Online reviews contain a body of various attitudes towards different items and places and help consumers understand what to expect. In this project, we employ distance metrics and clustering methods to conduct an exploration-based experiment on sentiment analysis of Yelp reviews dataset. Our goal is to uncover any underlying patterns and structures within the review corpus without relying on pre-existing linguistic annotations if possible. We show that with the variety of ratings and reviews, the clusters that form from the trained word vectors struggle to correctly classify - or find any pattern in - unannotated data enough to discern ratings.

1 Introduction

Clustering analysis stands as a fundamental technique in the realm of Natural Language Processing, enabling the exploration and organization of vast textual datasets without relying on predefined linguistic annotations. By grouping similar textual elements together, clustering facilitates various downstream applications such as topic modeling, sentiment analysis, and document summarization. Unlike supervised learning approaches that require labeled data for training, clustering methods operate in an unsupervised manner, making them particularly valuable for analyzing large corpora where annotations may be scarce or costly to obtain.

In this paper, we embark on an exploration-based experiment focused on analyzing a corpus of Yelp reviews using distance metrics and clustering methods. Yelp reviews represent a rich source of unstructured text data, offering insights into consumer opinions, preferences, and experiences across various businesses and services. Our goal is to leverage clustering techniques to uncover underlying patterns and structures within the Yelp review corpus without relying on pre-existing linguistic annotations.

By employing k-means clustering with word2vec features and hierarchical clustering with PCA, we aim to discern meaningful clusters of Yelp reviews based on content similarity and frequency of terms, respectively. These models serve as our primary tools for organizing and exploring the Yelp review dataset.

2 Corpus

The Yelp Reviews corpus was chosen as the primary dataset for this project due to several key attributes that make it particularly suitable for training and validation of an unsupervised experiment. The entirety of the corpus was 8 million reviews long; a sample of size 10,000 and 100,000 were used for training instances. The data in the corpus was a large `JSON` of form:

```
{review_id,user_id,business_id,stars,
useful,funny,cool,text,date}
```

This made the data easy to parse and select certain attributes for. For this experiment, the `stars` and `text` columns were used. Annotations the data had could be removed by simply selecting the `text` column and processing and training was done on that. `stars` was later compared against the labels generated by the K-Means model to determine performance. These features enhanced the ease-of-use and accessibility of experimentation as well as provide a method in case future changes need to be made with regards to rating classification or reception of the reviews.

3 Data

The problem at hand involves sentiment analysis in online reviews, one a rating scale rather than purely positive or negative. The objective of our supervised models is to accurately group different star ratings with each other, trying to be sim-

ilar the the rating provided on the Yelp data set. Data was taken from the Yelp Reviews Corpus and pre-processed by removing stopwords and non-alphanumeric symbols, tokenizing and lowercasing. Though there were various possible labels to pair with the `text` feature, `stars` was used as the label giving us a rating, review pair. Giving this information to a word2vec model created a list of vectors for the 33550 words in the vocabulary. Each sentence was from the corpus was then individually ran through the word2vec model, aggregated and normalized into a vector from that would represent the sentence as a whole. The vector representation of sentences in the corpus was converted into an array to be fed into K-Means.

4 Methods

For the experiment, we set up 2 models: a word2vec model for creating word vectors of elements found in the review as well as a K-Means model for identifying 5 groups of distinct ratings - reflecting the 5 star rating options. Furthermore, different parameters were used with each model using a **vector size** of 100 and **window size** of 5 as well as a **vector size** of 1000 and **window size** of 10. In our experiment, we utilized word2vec embeddings to represent the Yelp review corpus in a continuous vector space. Word2vec allowed sentence representations to capture the structure of the review by adding up the word vectors for each sentence element and taking the average. From the representations of the sentences, a K-Means model was trained allowing for meaningful comparisons between reviews based on their content. PCA was performed on the result of the K-Means grouping to plot the axes that would result in the most distinct groups. Finally, we find the ARI and Silhouette Score to see how well our unsupervised results aligned with the actual ratings.

5 Results

The results of the unsupervised clustering experiment, and the labels it produces were compared with the original star ratings from the data. The following graph in *Figure 1* shows the distribution of different cluster labels alongside the star labels from the Yelp Reviews dataset. The first trial was run with a vector size of 100 and window size of 5.

We see that the distribution of clustering labels compared to the distribution of truth labels yields no predictable pattern of any sort for most ratings.

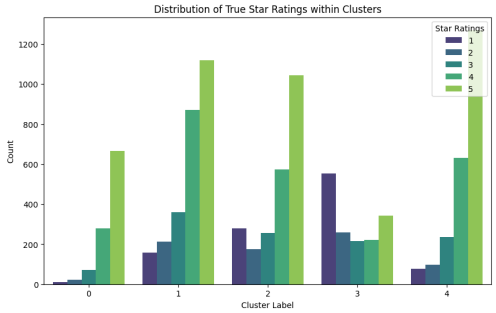

Figure 1: How true star rating are distributed among cluster ratings for a window of 5 and vector size of 100

In cluster labels *0, 1, 2* and *4*, the true stars given follow a similar pattern with most being assigned the *4* or *5* labels. Contrarily, cluster label *3* yields the highest amount of true label *1* and low values of everything else. This suggests that our model has capability to identify negative low star reviews but has limited capacity to cluster other reviews. With the high number of 5 star reviews in each cluster label, this suggests the model is also unable to distinguish what words are more positive.

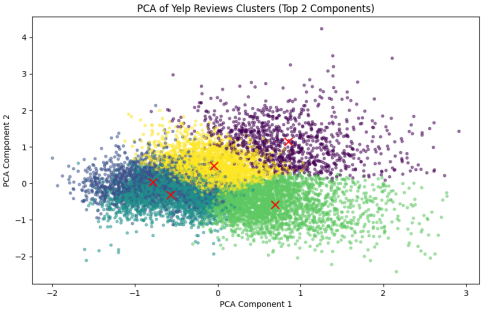

Figure 2: Visualization of clustering plotted with top 2 PCA components from sentences in Figure 1

Finding the 2 components that account for the most variance among the data using PCA and plotting the clusters, we find the groups seen in *Figure 2*. Rather than distinct groups of for each rating given for the reviews, all the vectors are clustered together. The similarity of vectors from each of the different cluster labels makes K-Means clustering from the vectors in ineffective way to determine the rating category of a review from its text.

We found that the Adjusted Random Index (ARI) for this data was 0.0304 and that the Silhouette Score was 0.1516. The score being close to 0 means that there is little similarity, neither positive nor negative, between the clustered labels and true labels. Likewise, a low Silhouette Score score

was also found, suggesting that data in each group had little similarity with each other. This pattern indicates that clusters were assigned close to randomly.

In the case that vector size, window size or data were no enough, the parameters were increased from 100 to 1000, 5 to 10 and 10,000 to 100,000 respectively. These changes resulted in the following distributions and clusters seen in *Figure 3* and *Figure 4*.

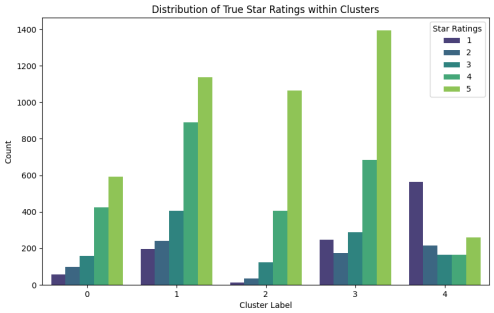

Figure 3: How true star rating are distributed among cluster ratings for a window of 10 and vector size of 1000

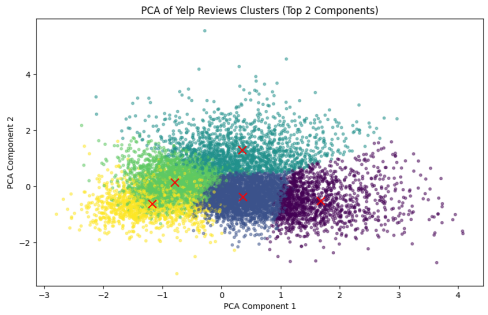

Figure 4: Visualization of clustering plotted with top 2 PCA components from sentences in Figure 1

The graphs above show similar trends to those found in *Figure 1* and *Figure 2*. The differences in vector size, window size and data were not meaningful nor impactful enough to improve the result of clustering.

6 Conclusion

Our experiment has provided significant evidence to back the claim that clustering the resulting data from word vectors is a poor method the detect the range of ratings found in online reviews. The consistent trend observed across different starting parameters highlights the difficulty in matching a review with a specific rating. Despite the diversity of the review dataset, our results indicate that the clusters formed from the trained word vectors struggle to accurately classify unannotated data, particularly in discerning sentiment patterns associated with different star ratings. Our findings suggest that using an unsupervised Word2Vec model alone for sentiment analysis lacks direct supervision for learning sentiment-related information, thereby resulting in suboptimal performance in capturing nuanced sentiment nuances accurately.

This task may prove difficult for even humans who are writing the reviews caused by interpretation differences and personal preference. While our model demonstrates the inability to group individual cluster ratings, it does show results in grouping specifically negative reviews suggesting there is a pattern in extremely negative cases. Despite these challenges, future research directions may involve exploring alternative feature representations, refining clustering algorithms, or incorporating additional contextual information to improve sentiment analysis accuracy.

Acknowledgements

We would like to express gratitude towards Dr. Dunn for his help and insights. We would also like to thank Yelp for making the review data publicly accessible for academic use.