Sentiment Analysis of Restaurant Reviews

Nitish Kumar(MT15041), Venkatesh(MT15016)

I. Introduction

A. Background

Zomato has been one of the most popular sites for users to rate and review local restaurants. Restaurants organise their own listings while users rate the business from 1 − 5 stars and write text reviews. Using this enormous amount of data that Zomato has collected over the years, it would be meaningful if we could learn to predict rating based on review's text alone, because free-text reviews are difficult for computer systems to understand, analyse and aggregate . The idea can be extended to many other applications where assessment has traditionally been in the format of text and assigning rating attached to it is difficult. Examples include predicting movie or book happiness based on news articles or blogs [2], assigning sentiment to YouTube videos based on viewers'comments, and even more general sentiment analysis, sometimes also referred to as opinion mining.

B. Goal and Outline

The goal of our project is to apply existing supervised learning algorithms to predict a review's rating on a given numerical scale based on text alone. We gathered data from Zomato . We experiment with different machine learning algorithms such as Naive Bayes, and BinarySVM and compare our predictions with the actual ratings. We develop our evaluation metric based on accuracy to quantitatively compare the effectiveness of these different algorithms. At the same time, we explore various feature selection algorithms such as using an existing sentiment dictionary, building our own feature set, removing stop words ,lemmatization, stemming.

C. Data

The data was scrapped from the Zomato website https://www.Zomato.com/ using Data Miner chrome addon .The Zomato dataset has information on reviews, users, businesses, and business check-ins.We specifically focus on reviews data that includes 345 user reviews of a single restaurant . We have only extracted text reviews and ignore the other information in the dataset for simplicity. We store the raw data in a cvs file . A higher rating implies a more positive emotion from the user towards the business. We randomly split this sample set into training (70% of the data) and test (the remaining 30%) sets for each class.

II. Results and Discussion

A. Evaluation Metric

We use Accuracy as the evaluation metric to measure our rating prediction performance. We compare our prediction with the actual rating to determine the effectiveness of our algorithm .

We received an overall accuracy of 70% using Binary SVM for the 'positive' and 'negative' classes .

B. Preprocessing

In our data preprocessing, we remove all the punctuations and all the multiple spaces from the review text. We convert all capital letters to lower case to reduce redundancy in subsequent feature selection. Cleaning up of unnecessary data like admin remarks was done in order to prevent mixing up of sentiment .

C. Feature Selection

We implement several feature selection algorithms, one using an existing opinion lexicon, the others building the feature dictionary using our training data with some additional variations . Our most basic feature selection algorithm uses Opinion Lexicon available for download publicly from http://www.unc.edu/~ncaren/haphazard/. This Opinion Lexicon is often used in mining and summarising customer reviews , so we consider it appropriate in our sentiment analysis. It consists of 6135 adjectives in total, where 2230 are positive, 3905 negative. We combine both the positive and negative words and define these words to be our features. The other feature selection algorithms loop over the training set word by word while building a dictionary that maps each word to frequency of occurrence in the training set. In addition, we implement some variations:

(1) Removing stop words (i.e. extremely common words) from the feature set using Terrier stop wordlist. (2) Stemming . (i.e. reducing a word to its stem/root form) to remove repetitive features using the Porter Algorithm readily implemented in Natural Language Toolkit (NLTK). (3) Lemmatisation (getting generic word for set of words) to combine words using WordNetLemmetizer readily implemented in the NLTK . [4] Vectorisation

We observe that building a dictionary from the dataset followed by removing stop words and stemming gives the highest prediction accuracy.The advantage of using an existing lexicon is that there is no looping over the dataset. Also, the feature set consists exclusively of adjectives that has sentiment meaning. The disadvantage is that the features that we use are not extracted from the Zomato dataset, so we might include irrelevant features while relevant features are not selected.

D. Naive Bayes

We use the Naive Bayes algorithm in the scikit-learn machine learning library to predict sentiment ratings. Similarly, the features are selected by looping over the training.set with stop words removed and Porter Stemming and Lemmatisation done .

Naive Bayes is traditionally used and proved to be the most suitable for text classification. In our Naive Bayes algorithm, we represent a review via a feature vector whose length is equal to the number of words in the dictionary.

E. SVM

We use the SVM algorithm in the scikit-learn machine learning library to predict sentiment ratings. Similarly, the features are selected by looping over the training.set with stop words removed and Porter Stemming and Lemmatisation done .

III. Conclusion and Future Work

In conclusion, we have experimented with various feature selection and supervised learning algorithms to predict sentiment ratings of the Zomato dataset using review text alone. We evaluate the effectiveness of different algorithms based on accuracy. We conclude that binarized Naive Bayes combined with feature selection with stop words removed and stemming is the best in our context of sentiment analysis.

Possible improvement could be extracting additional information from the dataset such as Business Categories and use customised feature sets for eachCategory, because different word features might be more or less relevant in different Business Categories.

Runtime of the algorithm could possibly be improved by training and testing within each business category, because of a smaller feature set. We could also try using parts-of-speech in feature selection process to differentiate between the same word features that are used as different parts-of-speech.

References

[1] G. Ganu, N. Elhadad, and A. Marian, “Beyond the Stars: Improving Rating Predictions using

Review Text Content.," WebDB, no. WebDB, pp. 1–6, 2009.

[2] N. Godbole, M. Srinivasaiah, and S. Skiena, "Large-Scale Sentiment Analysis for News and Blogs.,"ICWSM, 2007.

[3] Yun Xu, Xinhui Wu, Qinxia Wang"Sentiment Analysis of Zomato's Ratings Based on Text Reviews"