1. **Abstract**

Most consumers read online reviews before vising a business or using a service. The reviews not only provide customers with important information for making their decisions, but also help businesses get key insights from their customers and make improvements based on customer feedbacks. However, reading numerous reviews can be tedious and time-consuming. Therefore, it is important for reviews to have a specific class label to obtain useful information faster.

1. **Introduction**

In this project, a training data set that contains 56,000 Yelp reviews of businesses (such as restaurants, home services, auto services, etc.) is given. The reviews in the training are labeled as positive, negative or neutral. A classifier based on the training data is built and learnt by using Weka. A test set of 14,000 Yelp reviews without sentiment/class labels is also provided. The goal is to predict the sentiment/class labels for the 14,000 Yelp reviews by identifying critical attributes from the training data set and using data mining algorithms.

The report discuss the steps are made to preprocess the data and which WEKA built-in classification algorithm used.

1. **Data Preprocessing**
2. Emoji Process:

Some reviews contain emojis (i.e. :), :(, :D, etc.) that could be important in determining whether that review is positive, negative, or neutral. First, we created a python script to parse through the training data to find all the emojis that can be used in reviews. Then, we created a file named emoji2.xlsx containing all possible emojis and gave them a word and a score that will be used to replace it. After that, the script will get the total score of the all available emoji in a review that associated with it in the emoji file.

1. User Rating Process:

Many of the reviews have ratings that user rated in the review. These ratings are useful in classify the sentiment/class labels. Ratings are given in these formats: # stars, # out of #, # of #, #/#. We separate them into two parts, one is with the number ratings (e.g 5/10, 4 out of 5, etc) and another one is with the star ratings. Some people put multiple ratings for individual parts, so our script gets the average of all the ratings.

1. Punctuation Process:
2. Capitalized Words Process:
3. Sentiment Words Process:

Two data set contains positive and negative words that are commonly used in reviews are used to help with processing the words. All the positive and negative words are matched in each review with the two data set. The total number of positive and negative words are counted. The neutral words are the words those are not appeared in the positive and negative words data set, the frequency count of these words are added each time they appear. For example, if “okay” was found in 1000 neutral reviews, the score of neutral\_words is increased by 1000.

Text sentiments were used to get sentiment scores from the text. After the text are separated into individual words, the sentiment scores for each word are achieved by TextBlob.

1. Stop words:

Stop words are words that are commonly used that are not relevant to the goal that is trying to be achieved. Words such as pronouns (i.e. I, you, they, etc.), “a”, “the” are probably appear many times in all the reviews but those words are not useful to classify the sentiment/class labels. Therefore, it is needed to remove all words in the stop words file provided. After removing the stop words, the remaining words are filtered again by their frequency and only the top 10 number of words are kept.

1. Stemming:

Stemming is a method to get the base of an inflected word. That way, all forms of a word can be treated the same. For example, “eat” and “eating” can be treated as just “eat”. This allows us to group words faster since they would be transformed to the same word.

1. Dataset Cleaning:

There are a lot of meaningless special characters in the dataset, such as “#”, “\*”, “@”, etc. There are also some useless strings such as dates in the dataset. Therefore, regular expression is used to filter out of these characters.

1. Learning Algorithm
2. Appendix
3. Reference