**CSE 5334 - Data Mining**

**Semester Long Project – Finding the trending restaurants in a particular area using Yelp DataSet**

**Vidyadhar Angadiyavar** (1001103525) and **Rajesh Nayak** (1001126237)

**Objective**:

To make a list of top 10 trending restaurants in a particular location according to the user reviews for restaurants in Yelp. We expect to read the user reviews and then figure out whether a review is positive or negative, and also which of the aspects in a review (say, Food, Ambience, Kitchen, Service etc) have received a positive or negative rating.

Based on these reviews and their corresponding weight, we plan to calculate which 10 restaurants have got the highest score of positive reviews for a particular week. Week by week, the code is run to calculate the trending restaurants for that week.

**Challenges**:

1. Determining the different aspects of the review is quite difficult because it is hard to figure the user might have given a positive review for food and a negative one for ambience. To separate these aspects and to calculate an overall rating based on these different aspects will be challenging.
2. We need to perform sentiment analysis with a technique better than just using a ‘bag of words’. This is important because sentences like “I like this place… not!” and “The food is so good that I feel like saving it forever” will give a wrong rating if we use bag of words technique.
3. We are not sure about how to handle a small city with only few restaurants and fewer reviews for them. (Consider a scenario where more than 10 restaurants have only 1 or 2 ratings, it is quite tough to determine which one is better than the other)

**Methods and Techniques used/will be used:**

1. Perform sentiment analysis on every sentence in a review and classify them as positive and negative (We plan to use Naïve Bayes classifier in future to classify and predict the review sentiment)
2. Aspect based sentiment analysis will be used (<http://alt.qcri.org/semeval2015/task12/>) to calculate the review ratings for each aspect in a review. If the review is generic (does not have any particular aspect and is just the review of the restaurant, we will make a separate aspect called **General** for that purpose)
3. After having obtained the ratings for different aspects, we plan to calculate the weight of those ratings (because some restaurant might have only one review, which is a good review. So the percentage of good reviews is 100% for that restaurant. That does not completely make it the best restaurant in the locality) Thus, we will calculate the weight based on how many reviews the restaurant has received.
4. By calculating this weight, we will be able to obtain the top 10 trending restaurants by selecting the 10 restaurants with the highest weight score.

**Completed Tasks:**

1. We have read and understood how to go about the project and built a sentiment analysis algorithm using Naïve Bayes which ‘crudely’ classifies positive and negative reviews. We have used the train data provided by Kaggle (<http://inclass.kaggle.com/c/si650winter11>) currently, but we plan to use a bigger/better one later.
2. Built a website using PHP to enter a review and obtain the feedback whether the review is positive or negative on the browser.

**Yet to implement:**

1. Determine a better Sentiment analysis technique
2. Aspect Based sentiment analysis for calculating the rating for different aspects
3. Weight calculation

**Changes made to the project since the proposal:**

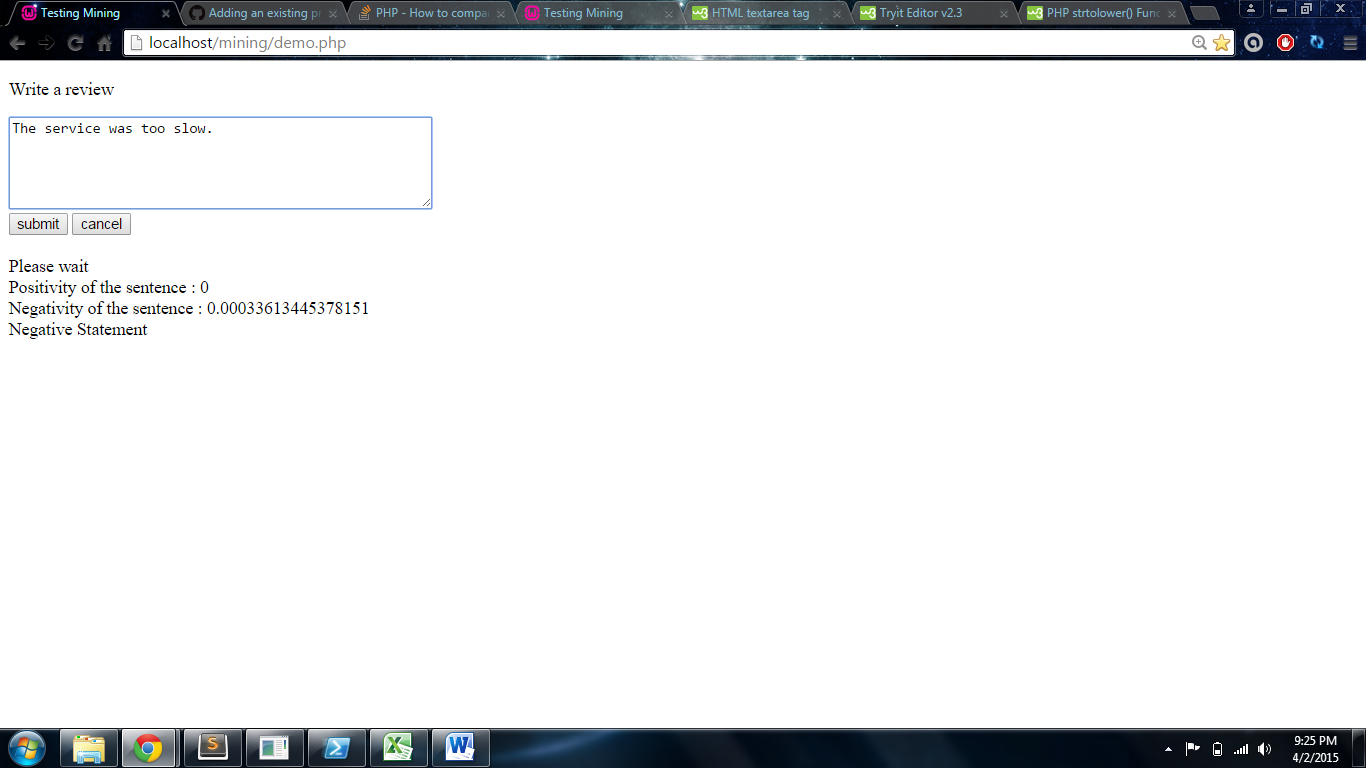
1. We were not aware of the Aspect Based Sentiment Analysis algorithm and hence we were unclear about how to find out the ratings of a particular aspect. Now we plan to implement this algorithm for that above mentioned purpose.

**Expected Challenges ahead:**

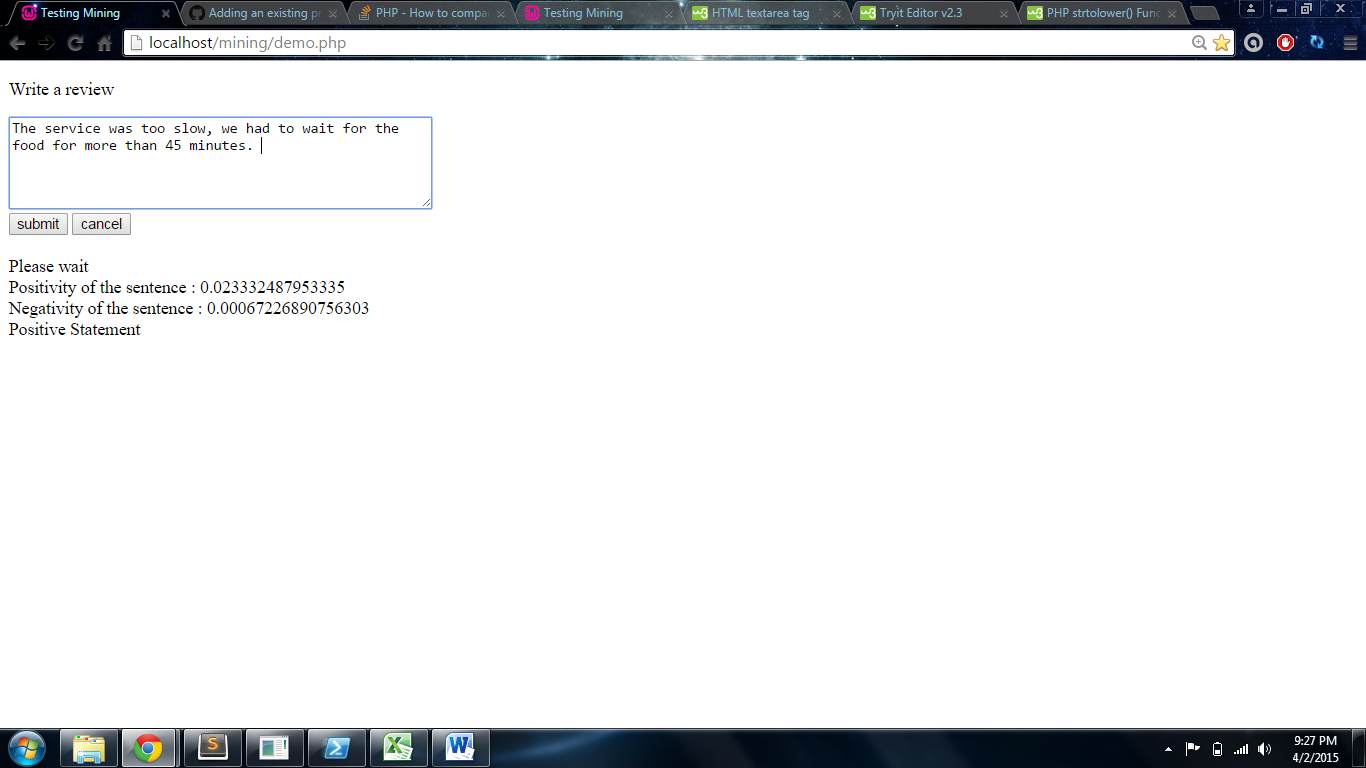
1. We are using ‘bag of words’ to perform sentiment analysis. This is just a basic implementation of the sentiment analysis because it just takes the count of positive and negative words and returns only either of these values. We need to implement a better technique for sentiment analysis to cover the sentences which have negation of negation scenarios, sarcasm, idioms and other variants.
2. We are yet to determine the technique to perform the above task, and it might consume a lot of time (depending on the technique) to implement it.

**Screenshots of the web page**

**For the next 2 screenshots, the code is working fine.** 



**Here, the word ‘wait’ has more probability for positive statements in the train data set than the negative statements. Hence we need to increase the number of entries in the train dataset and also change the algorithm to implement a better way of doing sentiment analysis than Naïve Bayes classifier.**



We also need to implement a method to identify the negation words (neither, nor, barely etc) and also to detect sarcasm. This is not working as of now, but we plan to implement it before the next submission.