Toronto Restaurants ranking by ethnicity using yelp data

# [Initial Results and Code Submission](https://courses.ryerson.ca/d2l/le/news/447411/622283/view)

**Data Cleaning:**

In this section we have done following exercises:

1. Upload the data (i.e. yelp. business & yelp.reviews)
2. Cleaning yelp.business data & eliminating unnecessary attributes
3. Cleaning yelp.reviews data & reduce the dimensionality.
4. Merging business & reviews files so we can have our desired datafile (i.e. Toronto Restaurants list & reviews).
5. Convert the json file to csv( i.e. yelp\_rev\_TorRes.csv)

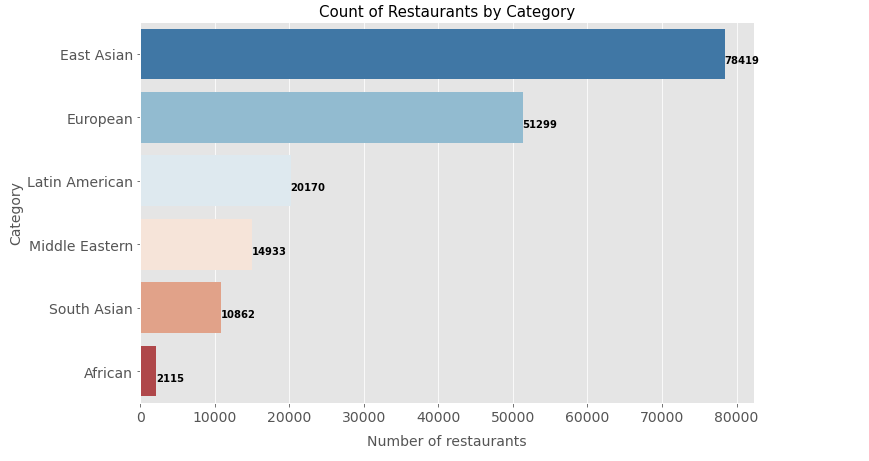
Made two dataset to avoid data lose.

1. Further cleaning of data.

**Exploratory data Analysis:**

In this part we have added Ethnicity with the help of categories.

we explore the ranking of the restaurants by ethnicity.



Then we classify our reviews rating into two classes of 0 (Negative for rating less than 3) and 1(Positive for rating >= 3) by creating new column as the "Label" for further processing.



Then we represent graphically as word cloud of frequently used words in the texts so we are going to investigate the frequency of words in the reviews.

Positive review word cloud we can indicate the frequency of occurrence of the word in the Positive reviews or compliments. We can see some interesting words like “quarantined” in positive reviews.

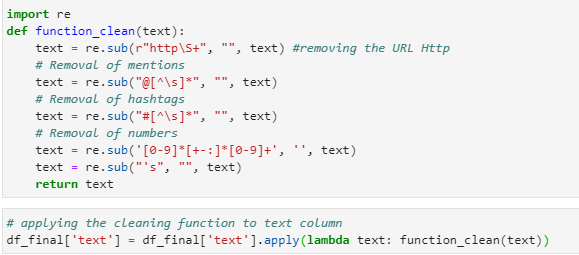


**Negative review word cloud using word cloud to show the frequency of words in negative reviews or complains.**



**Data Splitting – Train & Test Set:**

In order to train the classifier, we need to divide the dataset into train and test datasets. So we are going to split the reviews by 80:20 for train and test data. Before splitting the data, we create the function\_clean to clean up reviews.



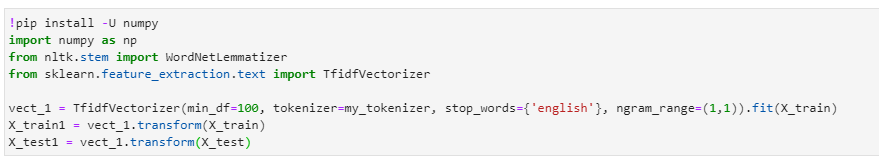


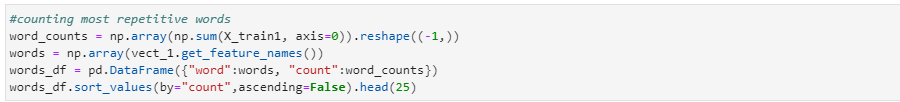
**Preprocessing Data:**

Now we are going to preprocess the reviews because all these modifications will directly affect the classifier’s performance. As we are going to use words as features so we can use some text formatting techniques which will help us in feature extraction including removing punctuation marks/digits ,and also stop-words. In addition, the implementation of lemmatization words using NLTK can be workable to maximize the performance. Tokenization is the last step to break reviews up into words and other meaningful tokens.

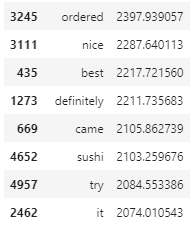
**By implementing the sklearn library, we can use TF\_IDF vectorizing to find the weighted words that occur more frequently in the document that leads to creation of the bag of words model. So our features will be the words or sequence of words of these reviews. We are going to explore different models with the combinations of n\_grams (unigrams,bigrams,trigrams).**

**Note: In this we explore only unigram.**



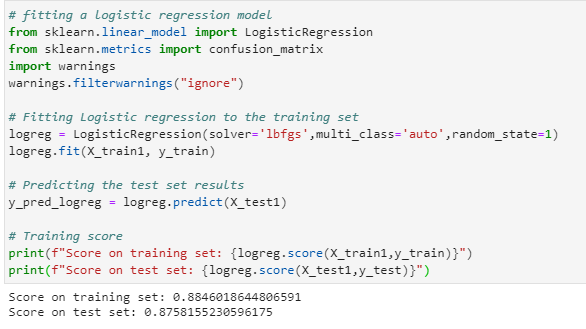


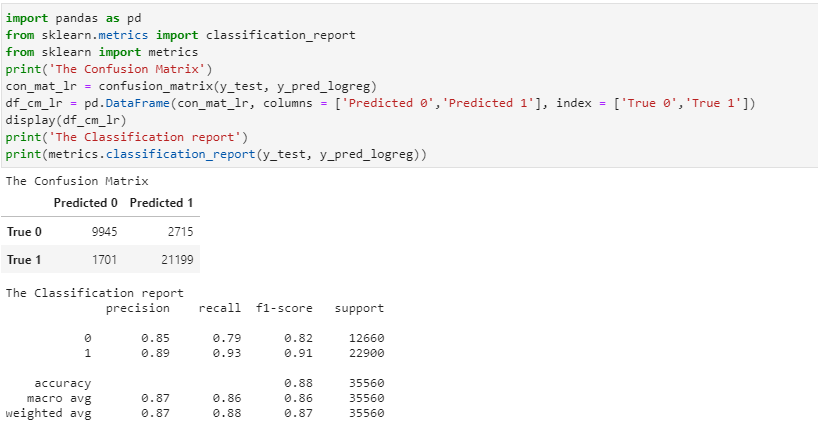




**Data Modeling:**

As the sentiment-classifier system has many applications from business to social sciences, the natural language processing and machine learning techniques are used to create the classifiers in order to explore the polarity of the reviews easily. Hence, we analyze the reviews given by the customers for the restaurant with the implementation of Logistic Regression.





Model Evaluation:

As shown above, our model test accuracy is 0.88. In order to evaluate the model further, we can extract some important evaluation metrics from the classification report such as precision, recall, and f1\_score. As shown in the above report, as the f1\_score is 0.91 for class 1 and 0.82 for class 0, therefore we can conclude this model is better in predicting positive reviews than negative ones.