Restaurant Rating Prediction

Final SMAI project

Varun Gupta, Vatsal Soni, Darshan Kansagara, Dhawal Jain

Problem Statement and Background :

Online reviews play a very important role in information dissemination and are influencing user decision. This project involved restaurant rating prediction on the basic of various factors that are useful for this purpose. For e.g Yelp dataset has features such as text review, tip information etc.

As proposed, we have implemented the model that can be used to predict the rating of the restaurant. Further used zomato API to extend the same problem to the indian scenario. We have used techniques like NLP for text reviews etc. to make it as exhaustive as possible.

Our data comes from the Yelp Dataset Challenge. As part of this challenge Yelp releases information about reviews, users and businesses from 4 US cities. The dataset (1.77 GB) is available for download on Yelp's contest page and contains the following information:

RAW DATA

Yelp Data :
yelp_academic_dataset_business.json
Yelp_academic_dataset_review.json

Zomato Data :
We used Zomato Web API, to fetch restaurant details, review details. We API auth key to fetch data and we are allowed to access only 1000 calls per day. Also we allowed to get top 100 restaurant list per city and 5 review per restaurant.So it was very challenging task to collect huge data for training model. We used multiple auth-key to get different small set of data and finally merge them all to get some data which is sufficient for training.

API used :

1) **To get list of Restaurant from city**
   https://developers.zomato.com/api/v2.1/search?entity_id=11&entity_type=city

2) **To get details of restaurant**
   https://developers.zomato.com/api/v2.1/restaurant?res_id=16774318

3) **To get details of review**
   https://developers.zomato.com/api/v2.1/reviews?res_id=16774318

IMPORTANT FEATURES

Business_id, name, text, review_count, stars, useful, is_open

Methods

Data Collection :

As mentioned above, we obtained the json dataset from Yelp. We converted these dataset into csv.
We used Zomato API to fetch zomato restaurant data and get review and restaurant csv.

Data Analysis

- Analyzed uniformity of data
- Analyzed most dominating words with respect to particular stars.
- Analyzed frequency of words
- Analyzed stars with respect to semantics
- Analyzed unwanted words

Data Cleaning

As part of data Cleaning following preprocessing is done

1. Remove record contains "NAN" value.
2. Remove extra white spaces from data.
3. Remove numeric values.
4. Remove punctuation mark.
5. Convert data to lowercase.
6. Remove unwanted words using stop words technique.
7. Reduce word to root form using stemming technique.

Data Transformation: Feature Extraction

Lexical features

Lexical features are traditionally the most relevant features in a text based model. As such we focused on extracting numerous lexical features. This extraction was memory intensive, and was performed on the EC2 instance. We stored these features in sparse matrix representation.
Lexical features were extracted after removing stop words.

● TFIDF:For tfidf features,we picked the 1000 most frequent words gathered from reviews and calculate their tfidf values.
● Unigrams of the 1000 most frequent words.
● 1000 most frequent bigrams in the data. After training SVMs on bigrams alone, we settled on using just the first 100 bigrams in our final model in the interest of time and performance.Examples:((u'go',u'back'),913),((u'first',u'time'),664),751),((u'really', u'good'), 636), ((u'great', u'place'), 600), ((u'ice', u'cream'), 491),etc.

Syntactic features

Syntactic features measure the part of speech distribution per review, i.e. the percentage of words that are verbs, nouns, adjectives, and adverbs.

Semantic features

We analyse positivity and negativity of review based on polarity of review, We also analysed mean polarity of a review corresponding to each star & thus formed a new feature semantic based on range of polarity.

Data Analysis: Modelling

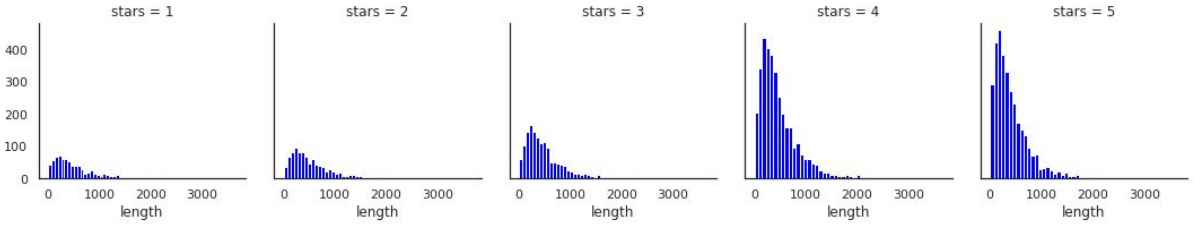
Following are the models on which we performed prediction task

- Logistic Regression
- Logistic Regression with one vs all
- Multinomial Naive Bayes Classifier
- Neural Network
- K-NN
- SVM

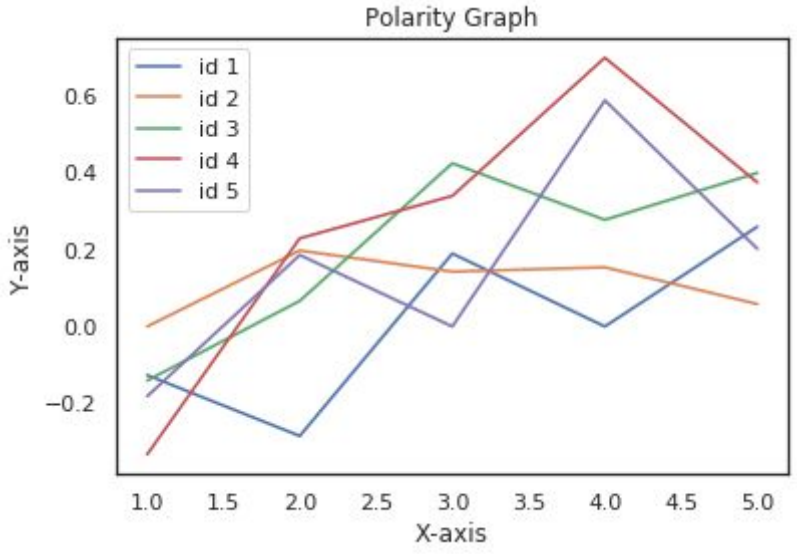
Data Visualization

1) **Word Cloud** : We created Word cloud to visualised the most common word and to remove useless word.

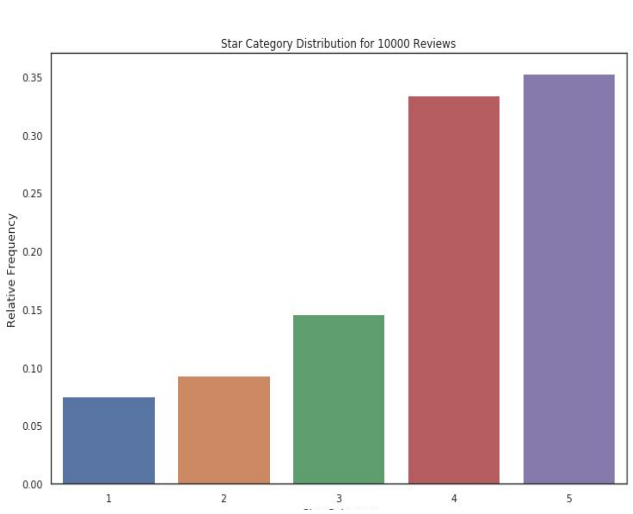
2) Review Length vs Stars

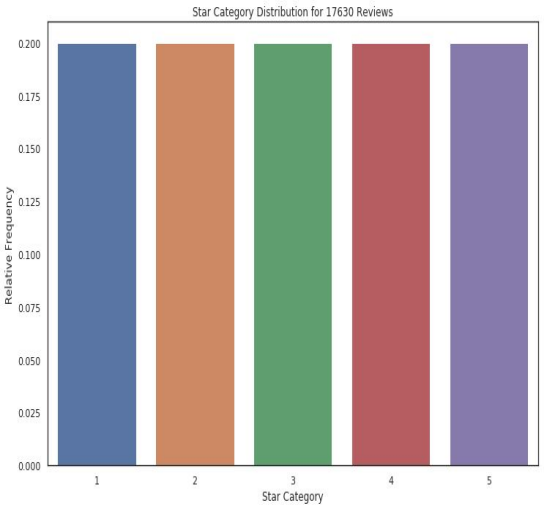


3) Polarity vs stars



4) Sampling and Upsampling





Results

Accuracy we get by applying following algorithms.

Logistic Regression :-	72.17%
Logistic Regression with one vs all :-	74.28%
Neural Network through multi-level:-	41.11%
Multinomial naive bayes :-	75.44%
KNN:-	83.19%
SVM :-	81.87%

Challenges :

- Collection of data from zomato API
- Convert large dataset from json file to csv file
- Data Analysis of the collected information for finding stop words, repeated structures.
- In zomato dataset, reviews are in hinglish (Hindi + English) form. So text processing and sentimental analysis becomes difficult.

Tools

NLTK: We used NLTK package for data cleaning and lexical feature extraction. The built in functions for removing stopwords and retrieving unigrams, bigrams were helpful.
The NLTK package worked well out of the box, but it was quite slow for POS tagging. We researched this topic for a fair amount of time, and came across the hunpos tagger. This combined with a model specifically meant for web data sped up our tagging process. highlight was how simple it was to use, and how it worked glitch free.

Scikit Learn: Standard implementations of ML models SVM, Logistic Regression, Logistic Regression one vs all, K-NN, Neural Network, Gradient boosting and Multinomial Naive Bayes.

Scikit learn performed reasonably well. Our SVM model on linear kernels took about an hour to converge on the complete dataset. But all other implementations were reasonably quick.

EXTENDED TO ZOMATO

Initially, we were using the same approach for sentiment analysis of the data collected using Zomato API as used for YELP dataset. As the ZOMATO dataset consists of English + Hindi text transliterated to English text, we were getting accuracy around ~50% . In order to improve so, we followed the below methodology. Due to time constraints, we were unable to implement the complete methodology but our aim is to complete it in near future.

The methodology consists of following phases:

1. Language Identification

Example:
Lol, itna bakwas food tha na who :)
lol, itna ghatiya dish that na woh. smile

English: dish, food , smile
Other (hindi): itna, tha, na, who, bakwas, ghatiya
Slang: lol = laugh out loud (with slang based sentiment scores)

2. Word Transliteration

After the word is identified to belong to a particular native language, then it is transliterated to the respective text, in our case, the Devanagari script.

3. Sentiment Scores Tagging

The transliteration obtained from the previous step is used in this step to find out the sentiment scores associated with the given word from the SentiWordNet available with us for both English and Hindi language.

4. Feature Extraction

Feature Generation is a time-consuming step that combines with all the above steps and gives a result, which is used as an input for machine learning based classification.

5. Supervised Learning Methods

Finally in this phase, we will get output as set of classified documents as per class label, which would be the sentiments of the document.

6. Output as sentiments of transliterated text documents

CONCLUSION

Based on all the analysis and experiments we would like to conclude that, review is the most dominating feature in deciding rating of the restaurant but through out the experiment and analysis phase of prediction with varieties of algorithm and combination of features K-NN comes out to be the most dominating algorithm in terms of accuracy of prediction.

TEAM CONTRIBUTION

Varun Gupta :-

- Pre-processing of data
  - Stop words
  - Stemming
  - Remove NAN values
  - Remove numeric data
  - Remove punctuation
  - Convert data to lowercase
  - Removing of meaningless data
- Extend it to zomato which handles English + Hindi transliterate English data.

Vatsal Soni :-

- Yelp data collection
- JSON to CSV converter
- Merge large dataset
- Sentiment analysis through polarity only
- Tuning Parameters
- Documentation

Darshan Kansagara :-

- Zomato Data collection from API
- Zomato and Yelp Data Analysis
- Using crawler to collect data from zomato web-site
- Documentation

Dhawal Jain :-

- Yelp data analysis
- CountVectorizer
- Pipelining
- Extracted TF-IDF , unigram and bigram features
- Modelled different models SVM, KNN, NN, Naive Bayes and Logistic Regression
- Prediction

REFERENCES

- https://developers.zomato.com/documentation
- https://www.kaggle.com/yelp-dataset/yelp-dataset
- https://elitedatascience.com/imbalanced-classes
- https://triton.ml/blog/tf-idf-from-scratch
- https://www.analyticsvidhya.com/blog/2018/02/natural-language-processing-for-beginners-using-textblob/
- https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7426090&tag=1
- https://medium.com/@zhiwei_zhang/final-blog-642fb9c7e781