Sentiment Analysis and Machine Learning on Yelp Reviews

Made by	NUID
Eric John Pozholiparambil	001811500
Rishi Raj Dutta	001825928

Abstract

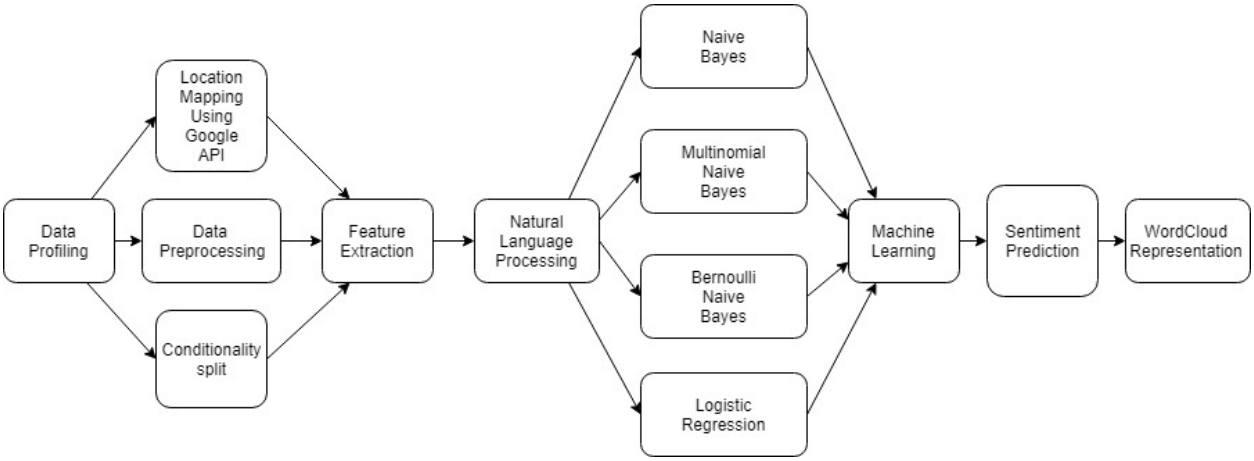
Sentiment analysis is one of the fastest growing research areas in computer science, making it challenging to keep track of all the activities in the area. Sentiment analysis is a series of methods, techniques, and tools about detecting and extracting subjective information, such as opinion and attitudes, from language. Millions of people use Yelp to find a good restaurant. Finding a restaurant depends on multiple aspects and various parameters such as services, popularity, accessibility, specialties. But the most important parameter on which a user makes a final decision is based on the reviews given by other users as customer experience is of utmost importance to businesses and customers. The main approach we have taken in this project is by using Naïve Bayes Classifier, Multinomial Naïve Bayes Classifier, Bernoulli Naïve Bayes Classifier and Logistic Regression to generate a Machine Learning Model to predict the sentiment value of a text on the basis of reviews generated by users

Introduction

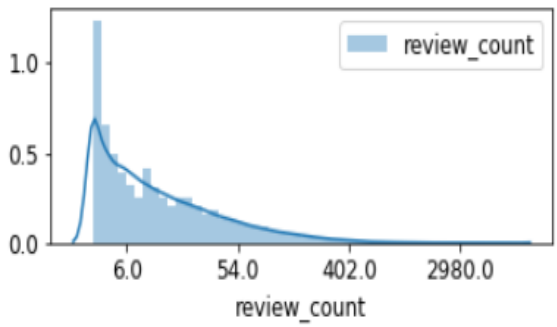
Microblogging websites have evolved to become a source of varied kind of information. This is due to nature of microblogs on which people post real time messages about their opinions on a variety of topics, discuss user experience, complain, and express positive sentiment for products and services they use in daily life. In fact, companies manufacturing such products have started to poll these microblogs to get a sense of general sentiment for their product. And Yelp serves as a platform to express the user's sentiment and opinion. With the increasing popularity of Yelp and the impact it incurs on the various businesses associated with Yelp due to the plethora of reviews. It is challenging to build a technology to detect and summarize an overall sentiment of all the reviews. We will take an approach to create a machine learning model based on the reviews in our dataset and then try to predict the sentiment of a review in real time based on the learning potential of the model.

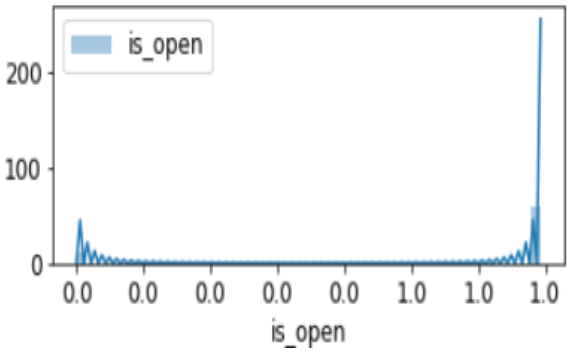
Code of Documentation

The Dataset that we have used is Yelp Dataset - A trove of reviews, businesses, users, tips, and check-in data from kaggle.com. The dataset contains six comma separator value files containing details of the users, businesses, business attributes, business hours, check-in, tips and reviews. The dataset contains about five million rows. Link to the dataset: https://www.kaggle.com/yelp-dataset/yelp-dataset. Below is a data model of the project flow which describes the steps taken to reach the sentiment prediction for the null values.



Initially we performed exploratory data analysis with the help of open python three environment tool kit. This approach generated distribution of all the numeric values that are there in the dataset as these numeric values were used in the classifier.





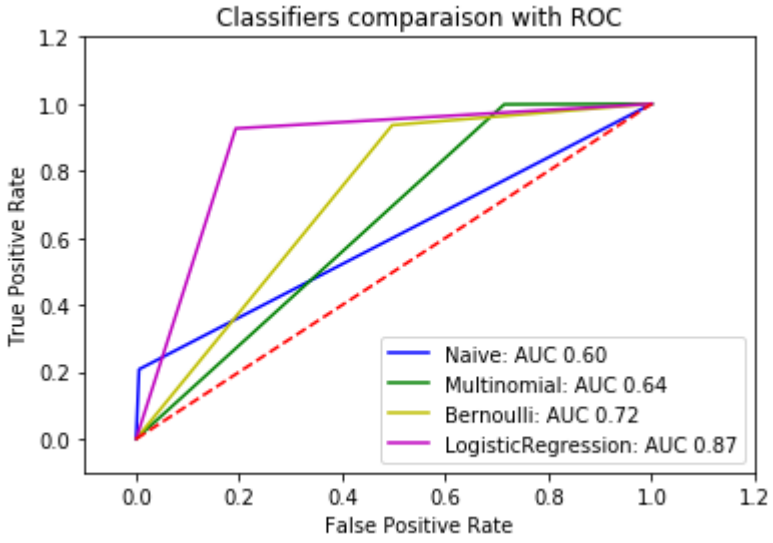
The above distribution gives us an understanding of all the users who are coming back again and giving their reviews over all the first-time new users because repeating users reviews create an increase and decrease in revenue and impacts the performance of the business.

Secondly, we have inserted a conditionality split on our reviews as defined, all the reviews which have received a star rating of three and below are clustered into a negative sentiment confidence and reviews which have received a star rating of four and five are clustered into positive sentiment confidence. This allows us to implement binary and Boolean operator classifier to train our machine learning model using Bernoulli Naïve Bayes Classifier. We have created dictionaries and stored them inside arrays in order to generate a sparse matrix for the implementation of Naïve Bayes classifier to train our machine learning model. These dictionaries are filled by extracting features of the reviews and indexed for iterating in the for loops**.**

Application of Multinomial Naive Bayes Classifier allows us to predict the sentiment of the rows containing null values for star rating. It divides the extracted feature words into individual tokens and calculates the occurrence count of each token. It then generates the sentiment result for each token. This classifier runs faster than Naive Bayes and works well for data which can easily be turned into counts, such as word counts in texts and also gives a higher sentiment accuracy model.

Results/Conclusion:

After running our model over four classifiers, which is Naïve Bayes, Multinomial Naïve Bayes, Bernoulli Naïve Bayes and Logistic Regression, on the test and train data, we found that Logistic Regression had a better accuracy as compared to the other classifiers by compromising on longer execution time.



In the above diagram, we can infer that Logistic Regression has a higher True Positive Rate and lower False Positive Rate as compared to Naïve Bayes which has lower True Positive Rate and a lower False Positive Rate.

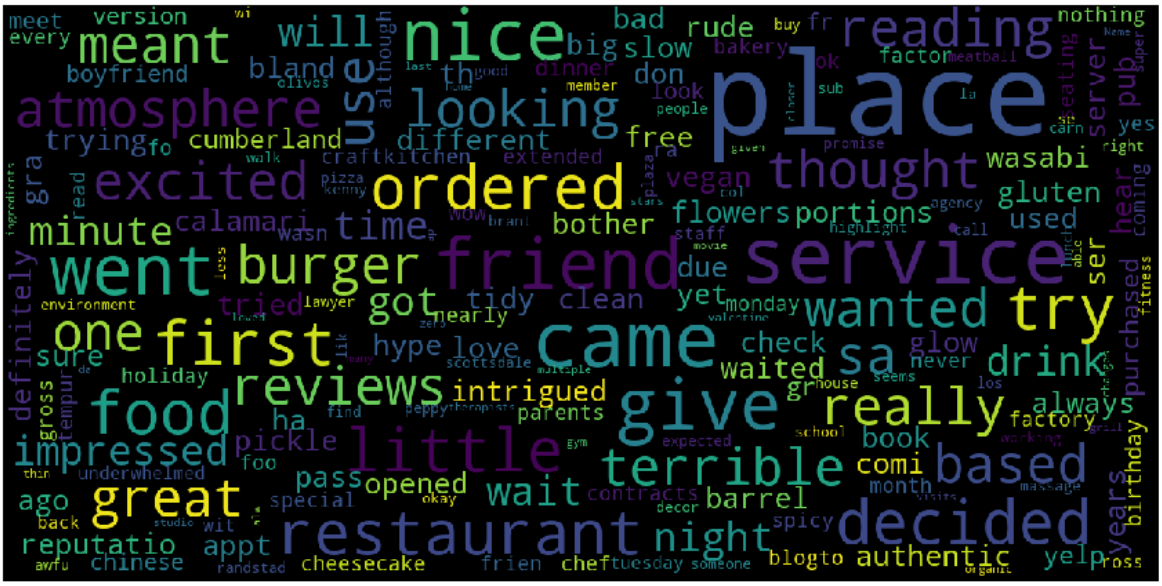
Out[79]:

	stars	text	user_id	Summary_Clean	words	Naive	multi	Bill	log
0	NaN	Love coming here. Yes the place always needs t...	bv2nCi5Qv5vroFiqKGopiw	love coming here yes the place always needs th...	[love, coming, here, yes, the, place, always, ...	neg	pos	pos	pos
1	NaN	Came here with my girlfriends one Sunday after...	u0LXt3Uea_GidxRW1xcsfg	came here with my girlfriends one sunday after...	[came, here, with, my, girlfriends, one, sunda...	neg	pos	pos	pos
2	NaN	worse customer service ever. \r\nManager on du...	u0LXt3Uea_GidxRW1xcsfg	worse customer service ever manager on duty wa...	[worse, customer, service, ever, manager, on, ...	neg	neg	neg	neg
3	NaN	Small little Japanese restaurant in the Don Mi...	u0LXt3Uea_GidxRW1xcsfg	small little japanese restaurant in the don mi...	[small, little, japanese, restaurant, in, the,...	neg	pos	pos	pos
4	NaN	Visiting from SF. Checked yelp and found this...	_L2SZSwf7A6YSrIHy_q4cw	visiting from sf checked yelp and found this p...	[visiting, from, sf, checked, yelp, and, found...	pos	pos	pos	pos
5	NaN	After being scared away from Rock & Rita's, we...	nOTI4aPC4tKHK35T3bNauQ	after being scared away from rock rita s we en...	[after, being, scared, away, from, rock, rita,...	neg	neg	neg	neg
6	NaN	So, below is is original review, which was acc...	nOTI4aPC4tKHK35T3bNauQ	so below is is original review which was accom...	[so, below, is, is, original, review, which, w...	neg	pos	neg	pos
7	NaN	I've visited this place on and off since the 7...	tL2pS5UOmN6aAOi3Z-qFGg	i ve visited this place on and off since the s...	[i, ve, visited, this, place, on, and, off, si...	neg	pos	pos	neg
8	NaN	Stopped in here yesterday for lunch. I wasn't ...	tL2pS5UOmN6aAOi3Z-qFGg	stopped in here yesterday for lunch i wasn t e...	[stopped, in, here, yesterday, for, lunch, i, ...	neg	pos	neg	neg
9	NaN	This is one huge casino. I've been in here man...	tL2pS5UOmN6aAOi3Z-qFGg	this is one huge casino i ve been in here many...	[this, is, one, huge, casino, i, ve, been, in,...	neg	pos	pos	pos

In the above diagram, we have defined the sentiments for each row having null values in their star ratings. Each Classifier is defined by a sentiment for the corresponding "Summary_Clean" review text.

Postive Words

In the above diagram, we have used Word cloud to represent the Positive Words from the text reviews in the given dataset.



Negitive words

In the above diagram, we have used Word cloud to represent the Negative Words from the text reviews in the given dataset.

Future Work:

We are currently working on extending the sentiment values from 2 (pos, neg) to 5 (very neg, neg, neutral, pos, very pos). We are trying to incorporate an extra classifier model against our model to see the comparison in the accuracy and loss of prediction, thereby improving the machine learning experience for the user. Also we would modify certain hyper-parameters within our model to see any significant difference within the accuracy and loss of our prediction.

References:

1. GoogleNet in Keras:

http://joelouismarino.github.io/blog_posts/blog_googlenet_keras.html

2. Going Deeper with Convolutions: https://www.cs.unc.edu/~wliu/papers/GoogLeNet.pdf

3. D. M. Freeman, “Using naive bayes to detect spammy names in social networks,” in Proceedings of the 2013 ACM workshop on Artificial intelligence and security. ACM, 2013, pp. 3–12.

4. How Naïve Bayes Classifier works: https://www.youtube.com/watch?v=XcwH9JGfZOU