Project Report

Sentimental Analysis on Restaurant Reviews

CSE 4022 – NATURAL LANGUAGE PROCESSING

Submitted by

SUBHASISH DAS (16BCE0435)
ANEESH KUMAR (16BCE0616)

Under the guidance of

Prof. SHARMILA BANU K.

Bachelor of Technology in
Computer Science and Engineering

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Computing Science and Engineering

April 2019

NLP Lab files:

https://github.com/subhasishdas/NLP-lab
ChatBot Code:
https://github.com/subhasishdas/ChatBot-Review-System-using-NLP-
Sentimental Analysis
https://github.com/subhasishdas/Sentimental-Analysis-using-NLP

1. Abstract

With a consistently growing middle class and increasing disposable income, the tourism and hospitality sector is witnessing a healthy growth and accounts for 7.5 per cent of the country's GDP. According to a report by KPMG, the hospitality sector in India is expected to grow at 16.1 per cent CAGR to reach Rs 2,796.9 thousand crore in 2022.The hospitality sector encompasses a wide variety of activities within the services sector and is a major job provider both direct and indirectly. The sector attracts the most FDI (Foreign Direct Investment) inflow and is the most important net foreign exchange earners for the country. It also contributes significantly to indirect tax revenue at the state and central level which includes revenues from VAT, Service Tax, and Luxury Tax etc.

Sentiment analysis of customer reviews has a crucial impact on a business's development strategy. Despite the fact that a repository of reviews evolves over time, sentiment analysis often relies on offline solutions where training data is collected before the model is built.

In this project will collect and analyze the reviews about the restaurants and helps us understand the people reaction and opinion towards that restaurant. Through this analysis we can generate the analysis report regarding the opinion of the people about a particular restaurant.

The purpose of this analysis is to build a prediction model to predict whether a review on the restaurant is positive or negative. To do so, we will work on Restaurant Review dataset, we will load it into predictive algorithms Multinomial Naive Bayes, Bernoulli Naive Bayes and Logistic Regression. In the end, we hope to find a "best" model for predicting the review's sentiment.

2. Problem Statement

The hotel industry in India is an important part of the hospitality and tourism infrastructure and a strategic part of India's growth story. Hotels are primarily viewed as a service industry with intangible areas of guest experience and service levels. The objective of this report is to better understand the hotel guest satisfaction and the areas that hotel management can change, in order to get better results. For this purpose, an analysis of hotel guest satisfaction ratings based on attributes such as Location, Sleep quality, Rooms, Service quality, Value for money and Cleanliness was performed. Further, text analysis of customer reviews was also performed to better understand the positive and negative sentiments of hotel guests. We focused on identifying the attributes that differentiate one hotel from another, and then using these attribute insights to make recommendation to hotel management, on how they can improve their operations, guest satisfaction and generally differentiate themselves from their competition. Data from an online website was used to analyse and compare customer ratings and reviews on five hotels. Statistical data analysis techniques

were used to identify the key attributes that are most important in choosing hotels and are critical to focus on in order to ensure guest satisfaction expectations are met. Based on text analytics, the key results from this study indicated that hotel guests look for a good room and a hotel with a pool and good service.

3. Introduction

What makes a good restaurant? What are the major concerns of customers for a great meal? Common knowledge may give general answers like delicious food, great services or pleasant environments, but they might not be true for different types of restaurants. In this project, we are going to unveil those essential features behind all kinds of restaurants via sentiment analysis on data

Businesses often want to know how customers think about the quality of their services in order to improve and make more profits. Restaurant goers may want to learn from others' experience using a variety of criteria such as food quality, service, ambience, discounts and worthiness

Users may post their reviews and ratings on businesses and services or simply express their thoughts on other reviews. Bad (negative) reviews from one's perspective may have an effect on potential customers in making decisions, e.g., a potential customer may cancel a service and persuade other do the same.

3. Literature Survey

No.	Author/Year	Title	Relevant Finding

1	M. GOVINDARAJAN Assistant Professor, Department of Computer Science and Engineering, Annamalai University, Annamalai Nagar, Tamil Nadu, India	Sentimental analysis on restaurant reviews using hybrid classification method.	In this research work, new hybrid classification method is proposed based on coupling classification methods using arcing classifier and their performances are analyzed in terms of accuracy. A Classifier ensemble was designed using Naïve Bayes (NB), Support Vector Machine (SVM) and Genetic Algorithm (GA). In the proposed work, a comparative study of the effectiveness of ensemble technique is made for sentiment classification. The feasibility and the benefits of the proposed approaches are demonstrated by means of restaurant review that is widely used in the field of sentiment classification. A wide range of comparative experiments are conducted about the effectiveness of ensemble technique for sentiment classification
2	Boya Yu, Jiaxu Zhou, Yi Zhang, Yunong Cao Center for Urban Science & Progress New York University	Identifying Restaurant Features via Sentiment Analysis on Yelp Reviews	Websites offers not enough information for independently judging its various aspects such as environment, service or flavor. In this paper, they have introduced a machine learning based method to characterize such aspects for particular types of restaurants. The main approach used in this paper is to use a support vector machine (SVM) model to decipher the sentiment tendency of each review from word frequency. Word scores generated from the SVM models are further processed into a polarity index indicating the significance of each word for special types of restaurant

3	I. Rish T.J. Watson Research Center	An empirical study of the naive Bayes classifier	The naive Bayes classifier greatly simplify learning by assuming that features are independent given class. Although independence is generally a poor assumption, in practice naive Bayes often competes well with more sophisticated classifiers. The broad goal is to understand the data characteristics which affect the performance of naive Bayes. Their approach uses Monte Carlo simulations that allow a systematic study of classification accuracy for several classes of randomly generated problems. They have analyzed the impact of the distribution entropy on the classification error, showing that low-entropy feature distributions yield good performance of naive Bayes
4	Tri Doan and Jugal Kalita University of Colorado Colorado Springs 1420 Austin Bluffs Pkwy, Colorado Springs	Sentiment Analysis of Restaurant Reviews on Yelp with Incremental Learning.	Sentiment analysis of customer reviews has a crucial impact on a business's development strategy. Despite the fact that a repository of reviews evolves over time, sentiment analysis often relies on offline solutions where training data is collected before the model is built. If we want to avoid retraining the entire model from time to time, incremental learning becomes the best alternative solution for this task. In this work, they present a variant of online random forests to perform sentiment analysis on customers' reviews. Their model is able to achieve accuracy similar to offline methods and comparable to other online models.

4. Tools and Methodologies

To build a model to predict if review is positive or negative, To do so, we have worked on the Restaurant Review dataset, we have loaded it into predicitve algorithms Multinomial Naive Bayes, Bernoulli Naive Bayes and Logistic Regression. In the end, we hope to find a "best" model for predicting the review's sentiment. following steps are performed.

4.1 Importing Dataset

Dataset: Restaurant_Reviews.tsv is a dataset from Kaggle datasets which consists of 1000 reviews on a restaurant.

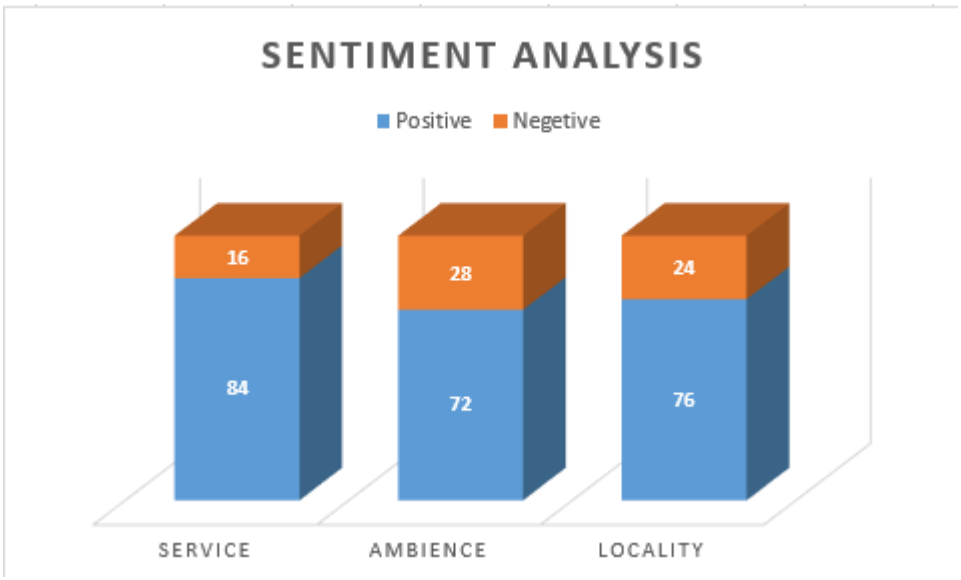
We have imported the libraries numpy and pandas for the project. NumPy is not another programming language but aPython extension module. It provides fast and efficient operations on arrays of homogeneous data.NumPy extends python into a high-level language for manipulating numerical data. Pandas is an open-source, BSDlicensed Python library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language.

4.2 Pre-processing Dataset

Each review undergoes through a pre-processing step, where all the vague information is removed like removing the stop words, numeric and special characters.

We have imported the library nltk and from nltk we have imported stopwords and Porter Stemmer for removing the stop words, numeric and special characters.

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries.



4.3 Vectorization

A way to represent text data for machine learning algorithm and the bag-of-words model helps us to achieve that task. The bag-of-words model is simple to understand and implement. It is a way of extracting features from the text for use in machine learning algorithms. The process of converting NLP text into numbers is called vectorization in ML

From the cleaned dataset, potential features are extracted and are converted to numerical format. The vectorization techniques are used to convert textual data to numerical format. Using vectorization, a matrix is created where each column represents a feature and each row represents an individual review.

sklearn.model_selection import train_test_split .ItSplit arrays or matrices into random train and test subsets

Quick utility that wraps input validation and next(ShuffleSplit().split(X, y)) and application to input data into a single call for splitting (and optionally subsampling) data in a oneliner.

4.4 Training and Classification

Further the data is split into training and testing set using Cross Validation technique. This data is used as input to classification algorithm.

Classification Algorithms:

Algorithms like Decision tree, Support Vector Machine, Logistic Regression, Naive Bayes were implemented and on comparing the evaluation metrics two of the algorithms gave better predictions than others.

1. Multinomial Naive Bayes

2. Bernoulli Naive Bayes

3. Logistic Regression

4.5 Analysis Conclusion

In this project, we have attempted to classify sentiment analysis for restaurant reviews using machine learning techniques. Two algorithms namely Multinomial Naive Bayes and Bernoulli Naive Bayes are implemented.

Flowchart

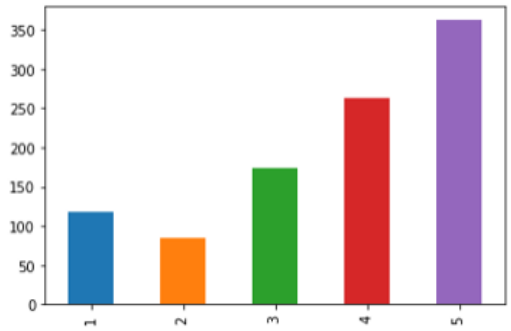
Analyzing the data

In [22]:
```
data['stars'].value_counts(sort=False)
```

Out[22]:
```
1    117
2     85
3    173
4    263
5    362
Name: stars, dtype: int64
```

In [23]:
```
data['stars'].value_counts(sort=False).plot.bar()
```

Out[23]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc068e982e8>
```



Assigning sentiments

We will be assigning positive sentiment to reviews having stars more than 3 and negative sentiment to reviews having stars less than 3. We will eliminate records having 3 stars

Let's remove all the reviews which contain 3 stars

In [24]:
```
data = data[data['stars'] != 3]
```

In [25]:
```
len(data)
```

Out[25]:
```
827
```

In [26]:
```
data.head()
```

Out[26]:

	review_id	user_id	business_id	stars	date	text	useful	funny	cool
0	x7mDIiDB3jEiPGPHOmDzyw	msQe1u7Z_XuqjGoqhB0J5g	iCQpiavjjPzJ5_3gPD5Ebg	2	2011-02-25	The pizza was okay. Not the best I've had. I p...	0	0	0
1	dDl8zu1vWPdKGihJrwQbpw	msQe1u7Z_XuqjGoqhB0J5g	pomGBqfbxcqPv14c3XH-ZQ	5	2012-11-13	I love this place! My fiance And I go here atl...	0	0	0
2	LZp4UX5zK3e-c5ZGSeo3kA	msQe1u7Z_XuqjGoqhB0J5g	jtQARsP6P-LbkyjbO1qNGg	1	2014-10-23	Terrible. Dry corn bread. Rib tips were all fa...	3	1	1
3	Er4NBWCmCD4nM8_p1GRdow	msQe1u7Z_XuqjGoqhB0J5g	elqbBhBfElMNSrjFqW3now	2	2011-02-25	Back in 2005-2007 this place was my FAVORITE t...	2	0	0
4	jsDu6QEJHbwP2Blom1PLCA	msQe1u7Z_XuqjGoqhB0J5g	Ums3gaP2qM3W1XcA5r6SsQ	5	2014-09-05	Delicious healthy food. The steak is amazing. ...	0	0	0

Let's assign positive or negative sentiments

In [27]:
```
data['sentiments'] = data['stars'] >= 4
```

In [28]:
```
data.head()
```

Out[28]:

	review_id	user_id	business_id	stars	date	text	useful	funny	cool	sentiments
0	x7mDIiDB3jEiPGPHOmDzyw	msQe1u7Z_XuqjGoqhB0J5g	iCQpiavjjPzJ5_3gPD5Ebg	2	2011-02-25	The pizza was okay. Not the best I've had. I p...	0	0	0	False
1	dDl8zu1vWPdKGihJrwQbpw	msQe1u7Z_XuqjGoqhB0J5g	pomGBqfbxcqPv14c3XH-ZQ	5	2012-11-13	I love this place! My fiance And I go here atl...	0	0	0	True
2	LZp4UX5zK3e-c5ZGSeo3kA	msQe1u7Z_XuqjGoqhB0J5g	jtQARsP6P-LbkyjbO1qNGg	1	2014-10-23	Terrible. Dry corn bread. Rib tips were all fa...	3	1	1	False
3	Er4NBWCmCD4nM8_p1GRdow	msQe1u7Z_XuqjGoqhB0J5g	elqbBhBfElMNSrjFqW3now	2	2011-02-25	Back in 2005-2007 this place was my FAVORITE t...	2	0	0	False
4	jsDu6QEJHbwP2Blom1PLCA	msQe1u7Z_XuqjGoqhB0J5g	Ums3gaP2qM3W1XcA5r6SsQ	5	2014-09-05	Delicious healthy food. The steak is amazing. ...	0	0	0	True

Replace True or False with 0 or 1

In [29]:
```
data['sentiments'] = data['sentiments'].astype(int)
```

In [30]:
```
data.head()
```

Out[30]:

	review_id	user_id	business_id	stars	date	text	useful	funny	cool	sentiments
0	x7mDIiDB3jEiPGPHOmDzyw	msQe1u7Z_XuqjGoqhB0J5g	iCQpiavjjPzJ5_3gPD5Ebg	2	2011-02-25	The pizza was okay. Not the best I've had. I p...	0	0	0	0
1	dDl8zu1vWPdKGihJrwQbpw	msQe1u7Z_XuqjGoqhB0J5g	pomGBqfbxcqPv14c3XH-ZQ	5	2012-11-13	I love this place! My fiance And I go here atl...	0	0	0	1
2	LZp4UX5zK3e-c5ZGSeo3kA	msQe1u7Z_XuqjGoqhB0J5g	jtQARsP6P-LbkyjbO1qNGg	1	2014-10-23	Terrible. Dry corn bread. Rib tips were all fa...	3	1	1	0
3	Er4NBWCmCD4nM8_p1GRdow	msQe1u7Z_XuqjGoqhB0J5g	elqbBhBfElMNSrjFqW3now	2	2011-02-25	Back in 2005-2007 this place was my FAVORITE t...	2	0	0	0
4	jsDu6QEJHbwP2Blom1PLCA	msQe1u7Z_XuqjGoqhB0J5g	Ums3gaP2qM3W1XcA5r6SsQ	5	2014-09-05	Delicious healthy food. The steak is amazing. ...	0	0	0	1

Preprocessing data

In [31]:
```
import re
import nltk
import string
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
filtered_text = []
for index, row in data.iterrows():
    review = re.sub('[^a-zA-Z]', ' ', row['text'])
    review = review.lower()
    review = review.split()
    ps = PorterStemmer()
    review = [ps.stem(word) for word in review if not word in set(stopwords.words('english'))]
    review = ' '.join(review)
    filtered_text.append(review)
```

In [32]:
```
filtered_text
```

```
 'back place favorit thai place ever go allll time never complaint start get known got busi servic start suck portion size g
ot cut half huge problem pay way less food last time went pork pad se ew tast good finish plate still hungri use know manag w
ould greet hello melissa nice see diet coke pad thai pad se ew day know still know disregard presenc also ask new portion siz
e answer great food worth money havent back year refus pay dinner still hungri sorri pinkaow use',
 'delici healthi food steak amaz fish pork awesom servic beyond bad thing say place worth everi penni',
 'place suck custom servic horribl dont serv food unless order pizza neighbor restaur dont control crowd mani time gone seen
fight bartend suck almost got fight one complet bitch refus serv drink busi celebr friend birthday behind bar place ridicul n
ever go ever',
 'like thai food tri origin thai bbq pad se ew die thai egg roll delici basil beef let menu anymor ask ye build fanci place y
                                                                                                    like thai food best town servic rock get gift cert restaur com beat t
                                                                                                    yth recommend everyon know know good thai food go',
 'amaz refer friend first thought korean mexican weird dont know husband area thought let tri friend noth great thing say pla
ce tri chimichanga absolut amaz actual chicken beef ever place make burrito half half havent meat fantast flavor beef girl or
```

Vectorization

In [33]:
```
# Creating the Bag of Words model using CountVectorizer

from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features = 1500)
X = cv.fit_transform(filtered_text).toarray()
y = data.iloc[:, -1].values
```

Training and Classification

Splitting the dataset into train and test

In [34]:
```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Logistic Regression Model

In [35]:
```
from sklearn.linear_model import LogisticRegression
logisticRegr = LogisticRegression()
logisticRegr.fit(x_train, y_train)
predictions = logisticRegr.predict(x_test)
```

Analyzing the result

In [36]:
```
# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, predictions)
print ("Confusion Matrix:\n",cm)
```

```
Confusion Matrix:
 [[ 22  21]
 [  5 118]]
```

In [37]:
```
# Accuracy, Precision and Recall
from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
score1 = accuracy_score(y_test, predictions)
score2 = precision_score(y_test, predictions)
score3= recall_score(y_test, predictions)
print("Accuracy is ",round(score1*100,2),"%")
print("Precision is ",round(score2,2))
print("Recall is ",round(score3,2))
```

```
Accuracy is  84.34 %
Precision is  0.85
Recall is  0.96
```

Multinomial Naive Bayes

In [38]:
```
# Fitting Naive Bayes to the Training set
from sklearn.naive_bayes import MultinomialNB
classifier = MultinomialNB(alpha=0.1)
classifier.fit(x_train, y_train)
```

Out[38]:
```
MultinomialNB(alpha=0.1, class_prior=None, fit_prior=True)
```

In [39]:
```
# Predicting the Test set results
y_pred = classifier.predict(x_test)
```

In [40]:
```
# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
print ("Confusion Matrix:\n",cm)
```

```
Confusion Matrix:
 [[ 25  18]
 [  6 117]]
```

Analyzing the result

In [41]:
```
# Accuracy, Precision and Recall
from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
score1 = accuracy_score(y_test,y_pred)
score2 = precision_score(y_test,y_pred)
score3= recall_score(y_test,y_pred)
print("\n")
print("Accuracy is ",round(score1*100,2),"%")
print("Precision is ",round(score2,2))
print("Recall is ",round(score3,2))
```

```
Accuracy is  85.54 %
Precision is  0.87
Recall is  0.95
```

Bernoulli Naive Bayes

In [42]:
```
# Fitting Naive Bayes to the Training set
from sklearn.naive_bayes import BernoulliNB
classifier = BernoulliNB(alpha=0.1)
classifier.fit(x_train, y_train)
```

Out[42]:
```
BernoulliNB(alpha=0.1, binarize=0.0, class_prior=None, fit_prior=True)
```

In [43]:
```
# Predicting the Test set results
y_pred = classifier.predict(x_test)
```

In [44]:
```
# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
print ("Confusion Matrix:\n",cm)
```

```
Confusion Matrix:
 [[ 25  18]
 [  5 118]]
```

Analyzing the result

In [45]:
```
# Accuracy, Precision and Recall
from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
score1 = accuracy_score(y_test,y_pred)
score2 = precision_score(y_test,y_pred)
score3= recall_score(y_test,y_pred)
print("\n")
print("Accuracy is ",round(score1*100,2),"%")
print("Precision is ",round(score2,2))
print("Recall is ",round(score3,2))
```

```
Accuracy is  86.14 %
Precision is  0.87
Recall is  0.96
```

Search specific reviews ¶

In [69]:
```
t = pd.read_csv('indexed_words.csv',error_bad_lines=False)
t = t.set_index('word')
t = t.iloc[:, 1:]
t.head()
# ast.literal_eval(t.loc['pizza']['review_id'])
```

Out[69]:

word	index	review_id
pizza	[0, 5, 124, 127, 129, 138, 140, 149, 150, 161,...	['x7mDIiDB3jEiPGPHOmDzyw', 'pfavA0hr3nyqO61oup...
okay	[0, 23, 108, 133, 135, 138, 261, 263, 296, 299...	['x7mDIiDB3jEiPGPHOmDzyw', 'MTrzrLQT_LK2VLK9xa...
best	[0, 6, 15, 20, 33, 37, 41, 42, 43, 63, 70, 103...	['x7mDIiDB3jEiPGPHOmDzyw', 'brokEno2n7s4vrwmmU...
i	[0, 3, 5, 6, 7, 9, 20, 23, 39, 41, 42, 46, 49,...	['x7mDIiDB3jEiPGPHOmDzyw', 'Er4NBWCmCD4nM8_p1G...
ve	[0, 5, 6, 9, 20, 39, 41, 52, 54, 65, 76, 82, 8...	['x7mDIiDB3jEiPGPHOmDzyw', 'pfavA0hr3nyqO61oup...

In [70]:
```
food = input('Enter food:')
```

```
Enter food:pizza
```

In [87]:
```
from nltk.corpus import stopwords
filtered_text = []
for i in rev['text']:
    review = re.sub('[^a-zA-Z]', ' ', i)
    review = review.lower()
    review = review.split()
    ps = PorterStemmer()
    review = [ps.stem(word) for word in review if not word in stopwords.words('english')]
    review = ' '.join(review)
    filtered_text.append(review)
```

In [109]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
vec = TfidfVectorizer(binary=True, use_idf=True)
tfidf_train_data = vec.fit_transform(filtered_text1)
tfidf_test_data = vec.transform(filtered_text)
```

In [115]:
```
x_train, x_test, y_train, y_test = train_test_split(tfidf_train_data, y, test_size=0.2, random_state=42)
classifier = BernoulliNB(alpha=0.7)
classifier.fit(x_train, y_train)
pred = classifier.predict(tfidf_test_data)
rev['pred'] = pred
```

Out[114]:

	text	pred
0	The pizza was okay. Not the best I've had. I p...	0
5	This place sucks. The customer service is horr...	1
124	Since it opened, this place has become a West ...	1
129	So lately I've been on a bit of an Italian kic...	0
140	I'm a sucker for top-notch pizza. By top-notch...	0
149	Walk in to Famoso's new Annex location and the...	0
150	La Passione is a dimly lit and intimately deco...	0
161	As soon as you step inside, your first thought...	1
162	Bar Salumi is the unassuming sister bar-come-r...	0
241	What is not to love about half-priced wine on ...	1
268	I grew up in Glendale and my parents would hav...	1
279	Dufferin Park is one of the top parks in Toron...	1
315	If I could give negative stars I would. Their...	1
429	Came from new York and this is by far the best...	1
447	Reminded me of an overpriced cheap frozen pizz...	1
502	On a Wednesday night, the restaurant was about...	1

4. **Experiment and Result**

We create the dataset using speech and reviews written by user. All the data is then pre-processed and trained using various algorithm to detemine the sentiment of the review given by the user.

In this study, we have made an attempt to classify sentiment analysis for restaurant reviews using machine learning techniques. Two algorithms namely Multinomial Naive Bayes and Bernoulli Naive Bayes are implemented.

Evaluation metrics used here are accuracy, precision and recall.

Using Multinomial Naive Bayes,

- Accuracy of prediction is 89.8%.
- Precision of prediction is 0.93
- Recall of prediction is 0.93

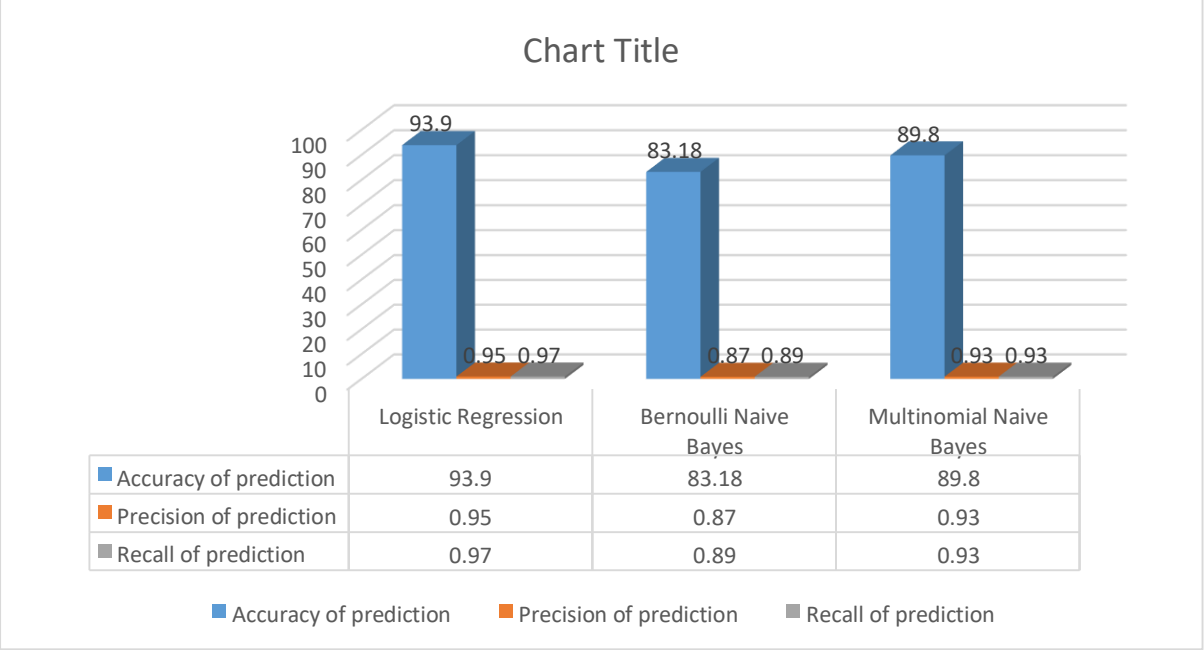
Using Bernoulli Naive Bayes,

- Accuracy of prediction is 83.18%.
- Precision of prediction is 0.87
- Recall of prediction is 0.89

Using Logistic Regression,

- Accuracy of prediction is 93.9%.
- Precision of prediction is 0.95
- Recall of prediction is 0.97

GRAPH



	Logistic Regression	Bernoulli Naive Bayes	Multinomial Naive Bayes
Accuracy of prediction	93.9	83.18	89.8
Precision of prediction	0.95	0.87	0.93
Recall of prediction	0.97	0.89	0.93

From the above results, Logistic Regression is slightly better method compared to Bernoulli Naive Bayes and Multinomial Naive Bayes, with 93.9% accuracy which means the model built for the prediction of sentiment of the restaurant review gives 93.9% right prediction.

6. Conclusion

In this project we have collected and analyze the reviews about the restaurants and helps us understand the people reaction and opinion towards that restaurant. Through this analysis we can generate the analysis report regarding the opinion of the people about a particular restaurant.

Through this report, many restaurants owners and chains a gets to know about the public opinion about their brand or their food quality and this help them to improve or analyze their food quality according to public opinion analyses.

Logistic Regression is slightly better method compared to Bernoulli Naive Bayes and Multinomial Naive Bayes, with 93.9% accuracy which means the model built for the prediction of sentiment of the restaurant review gives 93.9% right prediction

In marketing field, many brands use it to develop their strategies, to understand customers' feelings towards the food quality and the services of the restaurant.

7. References

[1] Sentiment Analysis of Restaurant Reviews on Yelp with Incremental Learning. Tri Doan and Jugal Kalita University of Colorado Colorado Springs 1420 Austin Bluffs Pkwy, Colorado Springs

[2] Sentimental analysis on restaurant reviews using hybrid classification method. M. GOVINDARAJAN Assistant Professor, Department of Computer Science and Engineering, Annamalai University, Annamalai Nagar, Tamil Nadu, India

[3] Identifying Restaurant Features via Sentiment Analysis on Yelp Reviews . Boya Yu, Jiaxu Zhou, Yi Zhang, Yunong Cao Center for Urban Science & Progress New York University

[4] Wikipedia

[5] An empirical study of the naive Bayes classifier I. Rish T.J. Watson Research Center