Capstone Project - 4

Unsupervised – ML – Zomato Restaurant Clustering and Sentiment Analysis

Sethupathy M

Contents

What is zomato?

Zomato is one of the most comprehensive and user-friendly apps where people can search for nearby restaurants and cafe's, order food online, and get it delivered at their doorstep in no time. Moreover, you can also get accurate information about restaurants as it provides menus, reviews, and ratings. Based on that, users can place orders and enjoy lip-smacking food at their homes.

Zomato was founded by **Deepinder Goyal** and **Pankaj Chaddah**, two **Delhi IIT** graduates, in **2008**. Till November 2010, Zomato was known as "**Foodiebay**." Once they saw their colleagues who were seeking menus of different restaurants to order food. That's when the idea took birth, and they thought of converting these manual menus into a digital format. In the year 2012, Zomato had spread its wings across the globe and started to list out the number of restaurants in the market.

How zomato works ?

The main work of Zomato is to suggest local and nearby restaurants to users and receive orders from them. Users can place orders from their favorite restaurant based on ratings and reviews shared by previous customers.

Step 1: From the desiccated app solution or website, users can explore various restaurants and order meals.

Step 2: Particular restaurant owners receive an order request and start preparing a meal.

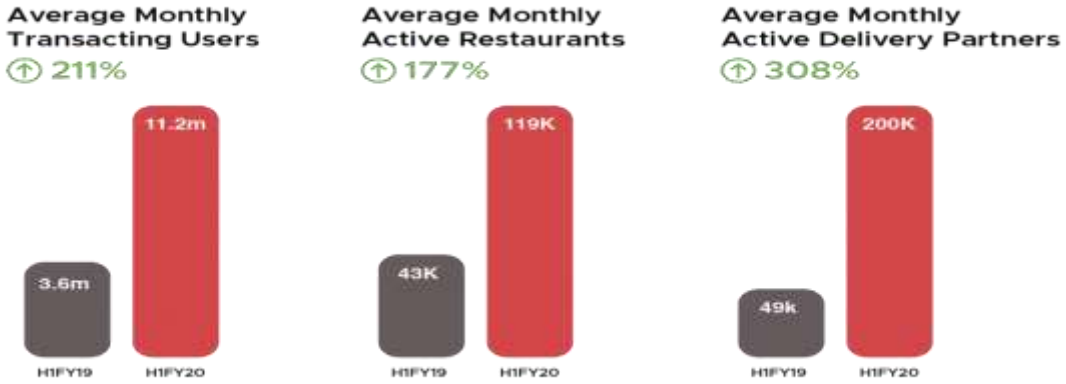
Step 3: Once the food is ready to dispatch, it will be handed over to delivery providers.

Step 4: Delivery providers deliver the meal to the customer's preferred location.

Step 5: From the given payment options, customers can make payments and share reviews based on their experience.

zomato Funds and Stats

Zomato received a total number of 909.6 million from different investors. Their funding was from Private Equity in 2020. Info Edge is a leading investor of Zomato. Other than that, Ant Financial, Delivery Hero, Shunwei Capital, Vy Capital, and many others are the investors of Zomato who have contributed their major stack to make Zomato popular worldwide. Now let's have a look at some interesting figures about Zomato's growth.



zomato Business Model

Zomato's Business Model is aimed at providing quality food services, information related to restaurants, their menus and user reviews. The Business Model of Zomato consists of providing food delivery services, information, user reviews and menu's of partner restaurants. It has created a revolution in industries doing food business by including different restaurants and facilitating people to look for restaurants more conveniently.

Addressing the Problem

The Project focuses on Customers and Company, and to analyze the sentiments of the reviews given by the customer in the data and made some useful conclusion in the form of Visualizations. Also, clustering the zomato restaurants into different segments. The data is visualized as it becomes easy to analyze data at instant. The Analysis also solve some of the business cases that can directly help the customers finding the Best restaurant in their locality and for the company to grow up and work on the fields they are currently lagging in.

The Project contains : Exploratory data analysis, Clustering, Sentiment Analysis which could help the customers to choose best restaurants and the restaurant owners to improve the restaurants in various aspects.

Features Summary

Zomato Restaurant Names and Metadata

- **Name –** Name of the Restaurants.
- **Links –** Links of the Restaurants.
- **Cost –** Average cost of the meal in Restaurants.
- **Collections** - The Collections in Zomato features popular restaurants across specific themes and trends at a particular location.
- **Cuisine –** A cuisine is specific set of cooking traditions and practices, often associated with a specific culture or region.
- **Timings –** Opening and closing time of the Restaurants.

Features Summary (continued)

Zomato Restaurant Reviews

- **Restaurants** – Name of the Restaurants.
- **Reviewer** – Name of the Reviewer.
- **Review –** Experience of the reviewer in the restaurant expressed in words.
- **Rating** - Experience of the reviewer in the restaurant expressed in numbers.
- **Metadata** – A set of data that describes and gives information about reviewers.
- **Time** – Time at which the reviews and ratings are given.
- **Pictures –** Number of pictures taken by the reviewer.

NULL value treatment

In the ‘Zomato Restaurant Names and Metadata’ dataset ‘Collections’ feature contains more than 50 % of its values as NULL. So ‘Collections’ feature is dropped.

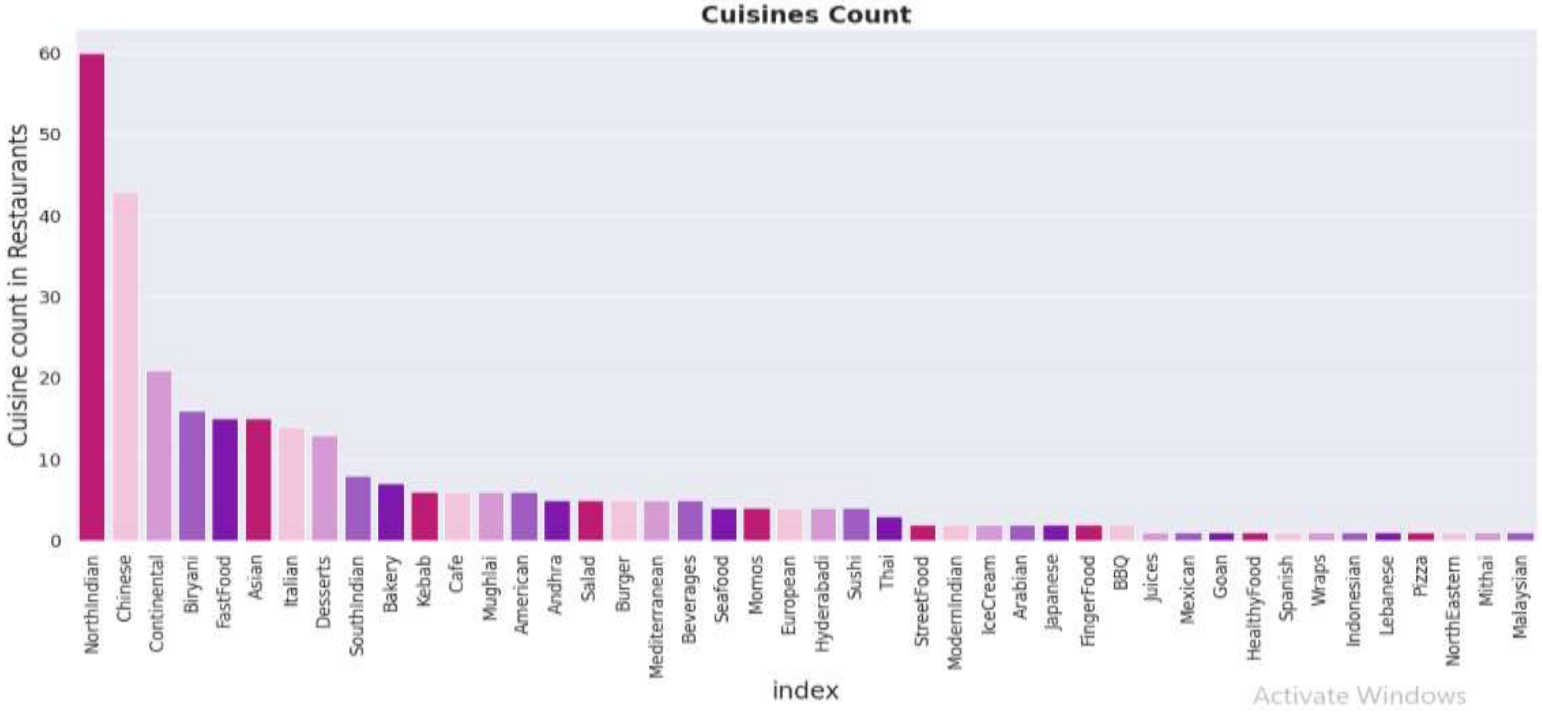
```
# Null values
# Percentage of null values for each features
# print(list(df.columns))

for col in list(df.columns):
  if ((df[col].isnull().sum())/(len(df[col]))*100) > 50:
    # print((df[col].isnull().sum())/(len(df[col]))*100)
    print('Feature with more than 50% of the observations are NULL values:',col)

# As in the feature 'Collections' more than 50% of the observations are NULL values, so feature 'Collections' is droppped
```

Exploratory Data Analysis

On **Zomato Restaurant Names and Metadata** dataset.

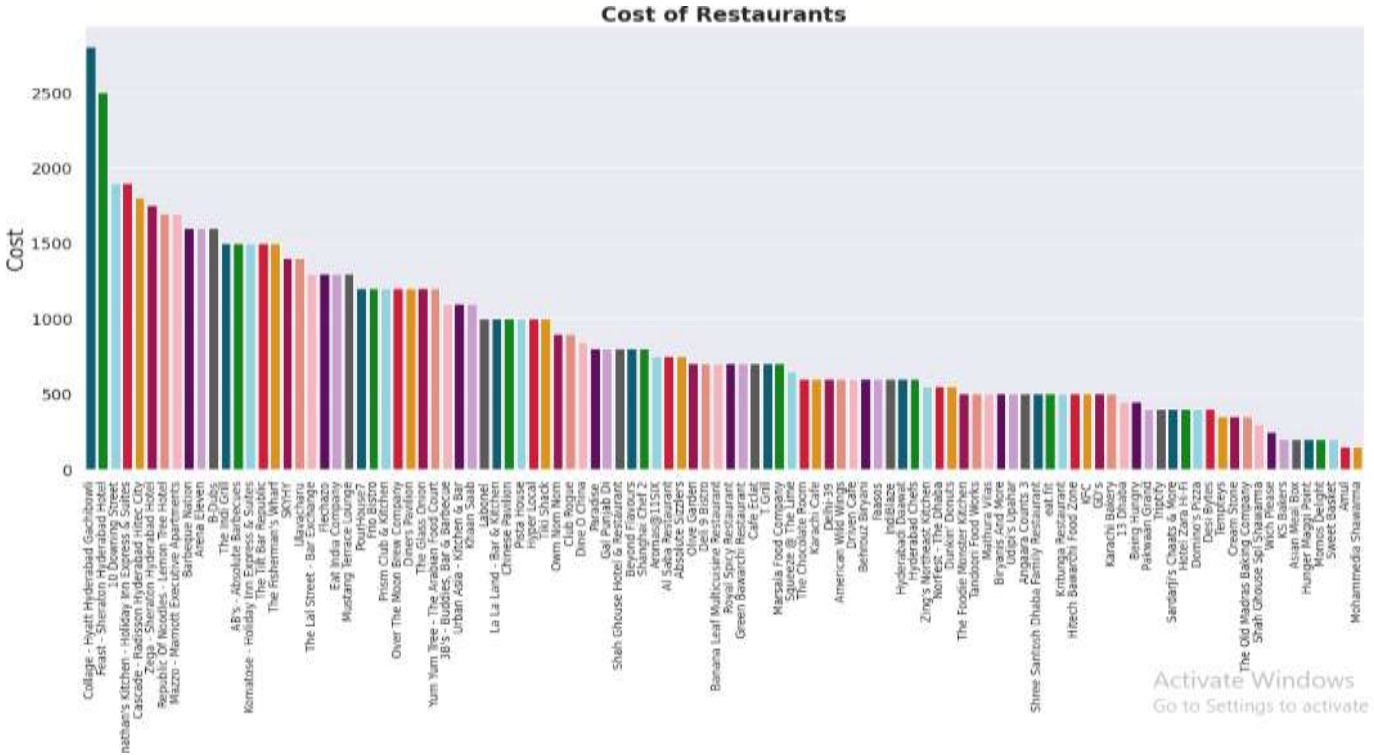


Exploratory Data Analysis (continued)

On **Zomato Restaurant Names and Metadata** dataset.

Exploratory Data Analysis (continued)

On **Zomato Restaurant Names and Metadata** dataset.



Text Preprocessing

On **Zomato Restaurant Names and Metadata** dataset.

Stemming

It is a technique used to extract the base form of the words by removing affixes from them. It is just like cutting down the branches of a tree to its stems.

Lemmatization

Lemmatization is a method responsible for grouping different inflected forms of words into the root form, having the same meaning.

```
def stem_words(words):
    """Stem words in list of tokenized words"""
    stemmer = LancasterStemmer()
    stems = []
    for word in words:
        stem = stemmer.stem(word)
        stems.append(stem)
    return stems

def lemmatize_verbs(words):
    """Lemmatize verbs in list of tokenized words"""
    lemmatizer = WordNetLemmatizer()
    lemmas = []
    for word in words:
        lemma = lemmatizer.lemmatize(word, pos='v')
        lemmas.append(lemma)
    return lemmas

def normalize(words):
    words = remove_non_ascii(words)
    words = to_lowercase(words)
    words = remove_punctuation(words)
    words = replace_numbers(words)
    return words
```

Text Preprocessing (continued)

On **Zomato Restaurant Names and Metadata** dataset.

- **Non-ASCII** characters are those that are not encoded in ASCII, such as Unicode,EBCDIC..
- Converting Uppercase letters to **Lowercase**.
- **Removing punctuations**.

```
def remove_non_ascii(words):
    """Remove non-ASCII characters from list of tokenized words"""
    new_words = []
    for word in words:
        new_word = unicodedata.normalize('NFKD', word).encode('ascii', 'ignore').decode('utf-8', 'ignore')
        new_words.append(new_word)
    return new_words

def to_lowercase(words):
    """Convert all characters to lowercase from list of tokenized words"""
    new_words = []
    for word in words:
        new_word = word.lower()
        new_words.append(new_word)
    return new_words

def remove_punctuation(words):
    """Remove punctuation from list of tokenized words"""
    new_words = []
    for word in words:
        new_word = re.sub(r'[^\w\s]', '', word)
        if new_word != '':
            new_words.append(new_word)
    return new_words

def replace_numbers(words):
    """Replace all interger occurrences in list of tokenized words with textual representation"""
    p = inflect.engine()
    new_words = []
    for word in words:
        if word.isdigit():
            new_word = p.number_to_words(word)
            new_words.append(new_word)
        else:
            new_words.append(word)
    return new_words
```

Text Preprocessing (continued)

On **Zomato Restaurant Names and Metadata** dataset.

TFIDF vectorizer

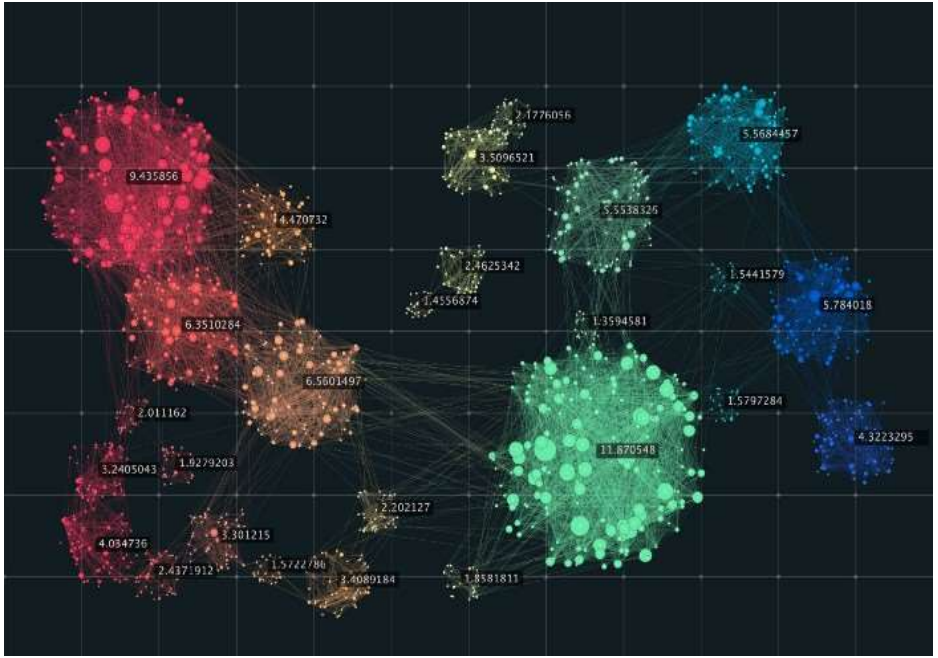
Term frequency-inverse document frequency is a text vectorizer that transforms the text into a usable vector. It combines 2 concepts, Term Frequency (TF) and Document Frequency (DF). The term frequency is the number of occurrences of a specific term in a document.

```
# Tfidf vectorizer
vectorizer = TfidfVectorizer(stop_words= 'english')
X = vectorizer.fit_transform(df['Cuisines'])
```

Clustering

On **Zomato Restaurant Names and Metadata** dataset.

Clustering – It is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group than those in other groups.

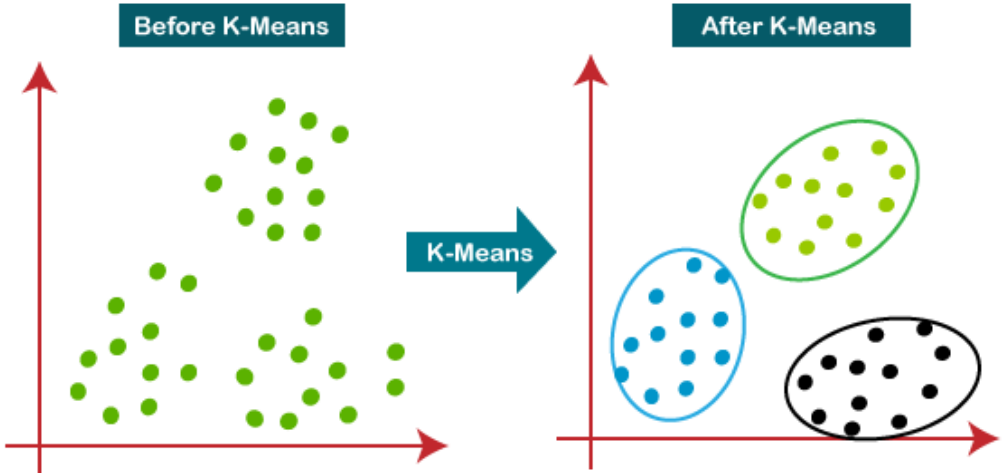


Clustering (continued)

On **Zomato Restaurant Names and Metadata** dataset.

K Means Clustering

The K-means clustering algorithm computes centroids and repeats until the optimal centroid is found. It is presumptively known how many clusters there are. It is also known as the flat clustering algorithm. The number of clusters found from data by the method is denoted by the letter 'K' in K-means.

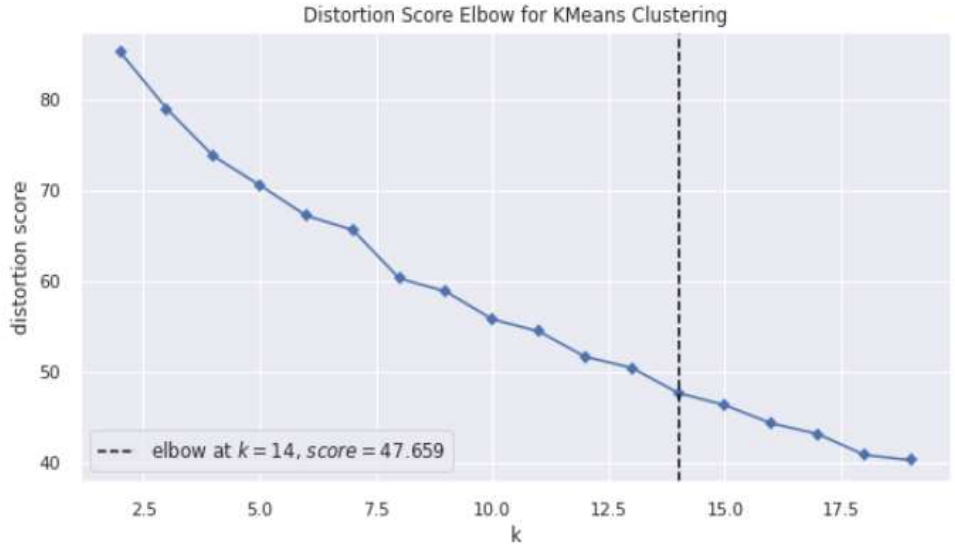


Clustering (continued)

On **Zomato Restaurant Names and Metadata** dataset.

Finding appropriate ‘K’ value

```
# Function to find appropriate 'K' value
def KElbowvisualizer(metric):
  model = KMeans(init="k-means++",max_iter=300,random_state=0)
  plt.figure(figsize=(10,5))
  sns.set(font_scale = 1)
  visualizer = KElbowVisualizer(model, k=(2,20),metric= metric,
  # plt.title(fontweight='bold')
  # # Fit the data to the visualize
  visualizer.fit(X)
  visualizer.poof()
```



Clustering (continued)

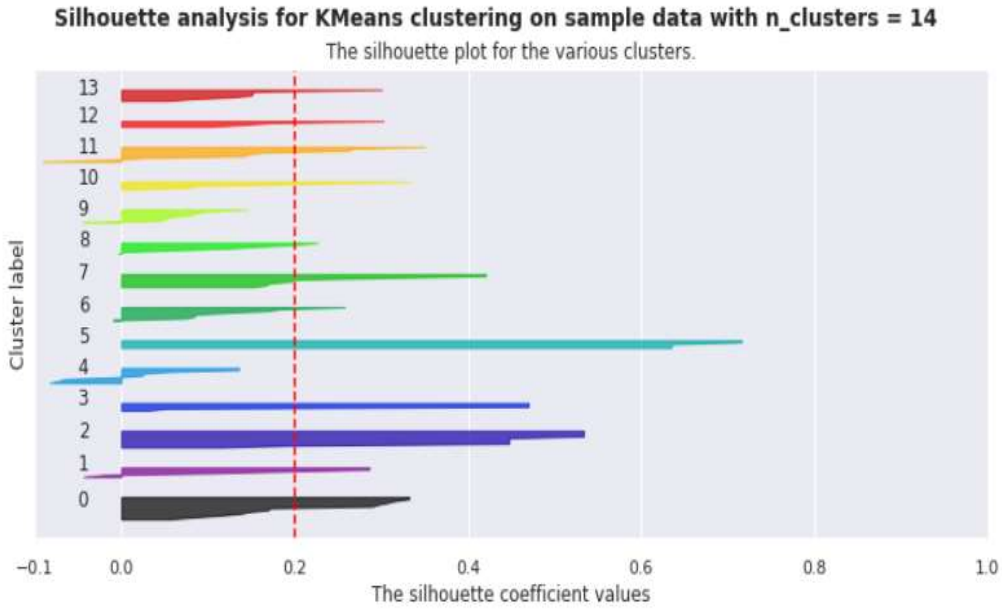
On Zomato Restaurant Names and Metadata dataset.

- **Working of K-Means Algorithm**
- The following stages will help us understand how the K-Means clustering technique works-
- **Step 1:** First, we need to provide the number of clusters, K, that need to be generated by this algorithm.
- **Step 2:** Next, choose K data points at random and assign each to a cluster. Briefly, categorize the data based on the number of data points.
- **Step 3**: The cluster centroids will now be computed.
- **Step 4:** Iterate the steps below until we find the ideal centroid, which is the assigning of data points to clusters that do not vary.

Clustering (continued)

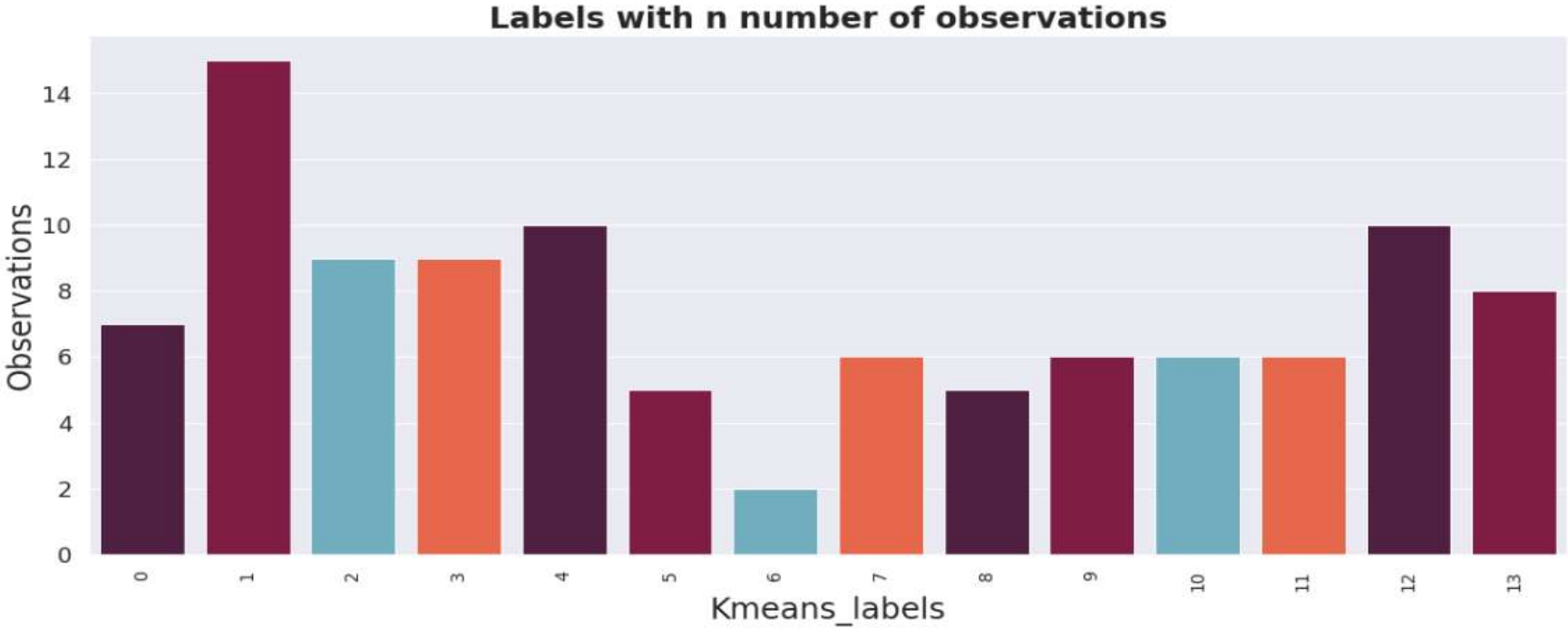
On Zomato Restaurant Names and Metadata dataset.
K Means Clustering Model Validation

```
For n_clusters = 10 The average silhouette_score is : 0.1825
For n_clusters = 11 The average silhouette_score is : 0.1747
For n_clusters = 12 The average silhouette_score is : 0.1937
For n_clusters = 13 The average silhouette_score is : 0.1866
For n_clusters = 14 The average silhouette_score is : 0.1998
For n_clusters = 15 The average silhouette_score is : 0.1945
```



Clustering (continued)

On Zomato Restaurant Names and Metadata dataset.
K Means Clustering

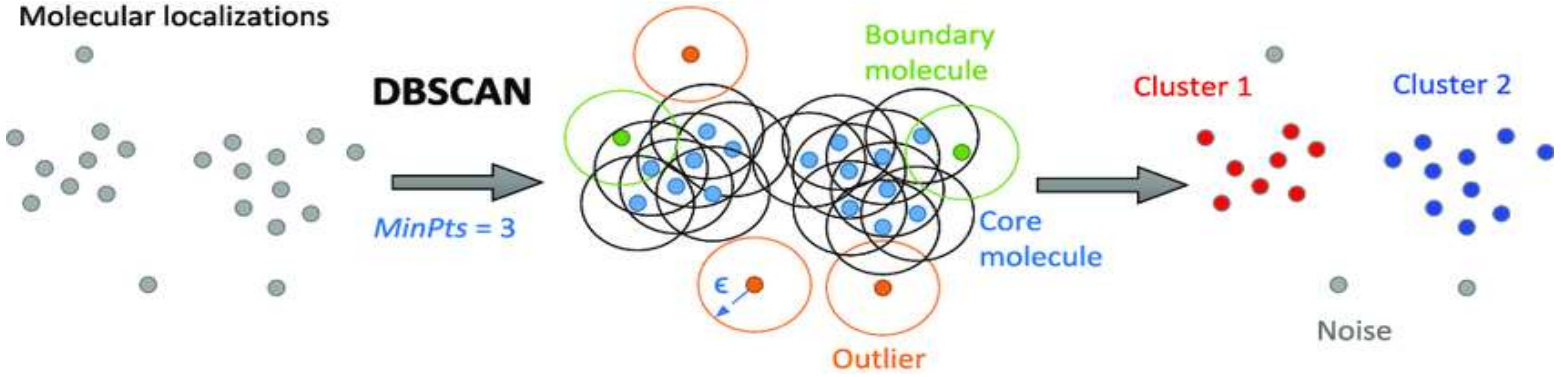


Clustering (continued)

On Zomato Restaurant Names and Metadata dataset.

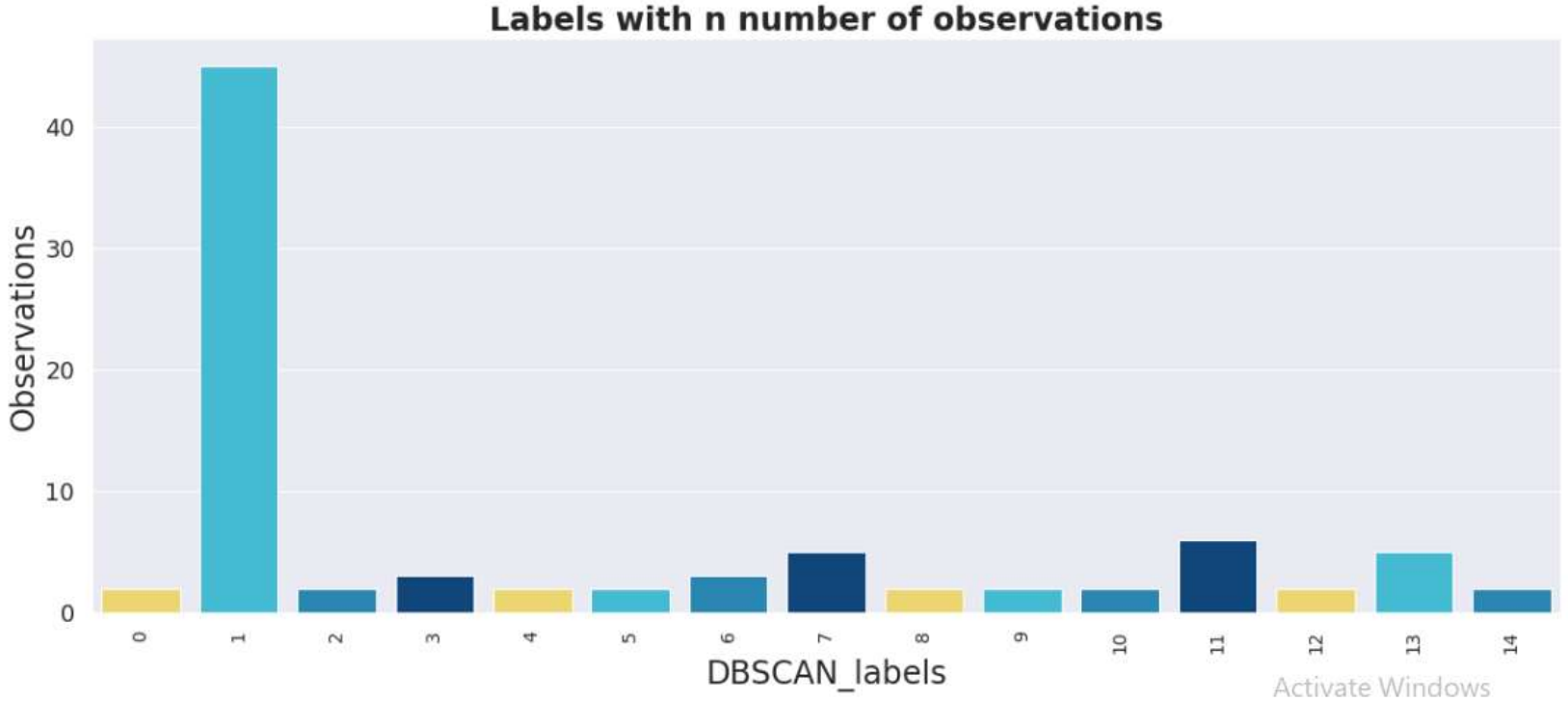
DBSCAN Clustering

DBSCAN is a density-based clustering algorithm that works on the assumption that clusters are dense regions in space separated by regions of lower density. It groups 'densely grouped' data points into a single cluster. It can identify clusters in large spatial datasets by looking at the local density of the data points. The most exciting feature of DBSCAN clustering is that it is robust to outliers.



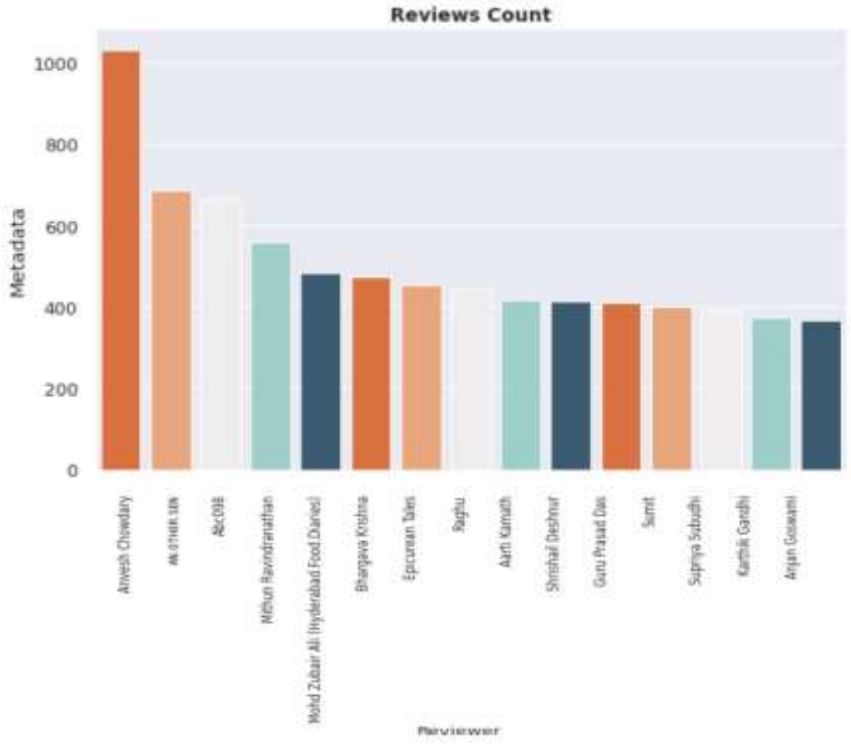
Clustering (continued)

On Zomato Restaurant Names and Metadata dataset.
DBSCAN Clustering



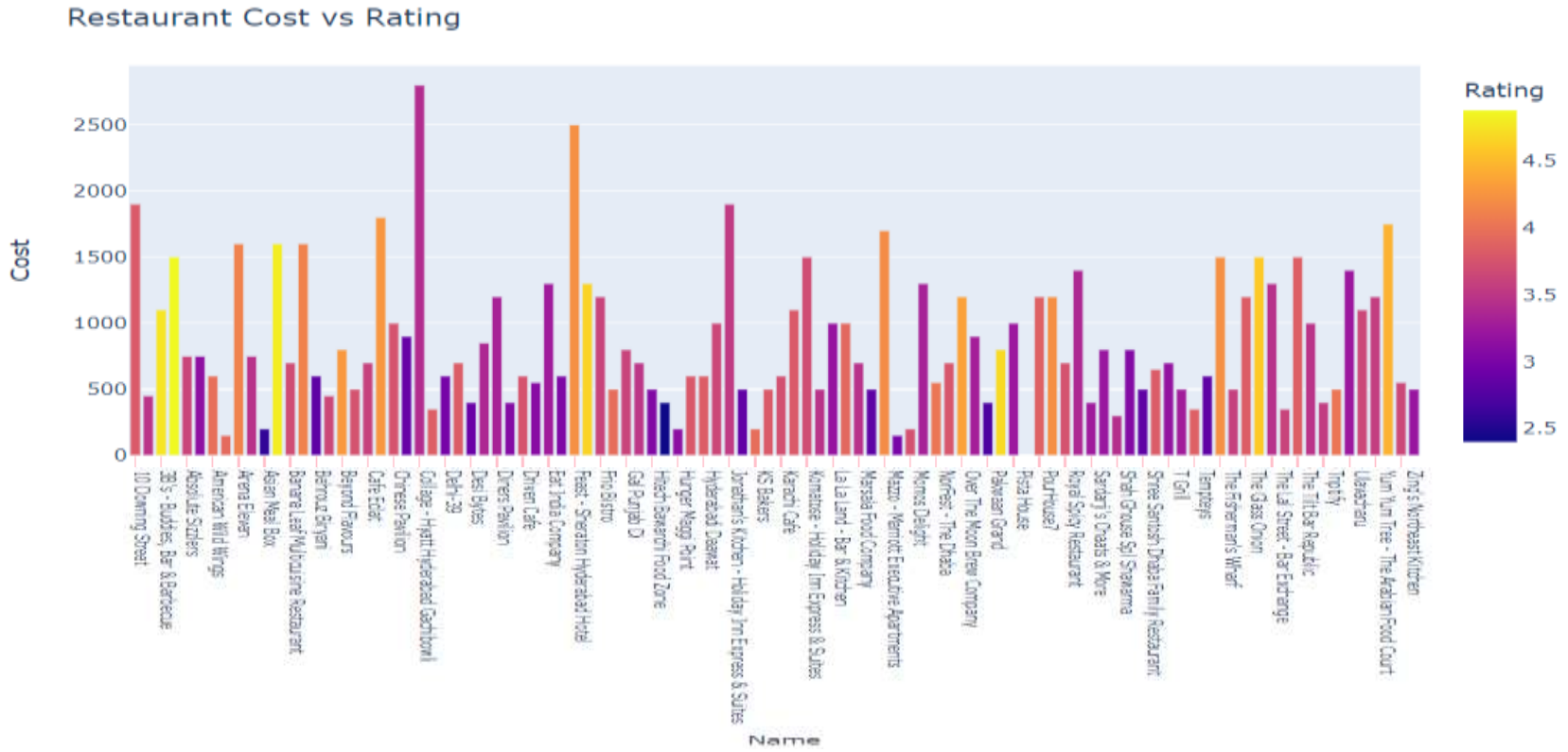
Exploratory Data Analysis

On Zomato Restaurant Reviews dataset.



Exploratory Data Analysis (continued)

On Merged dataset.



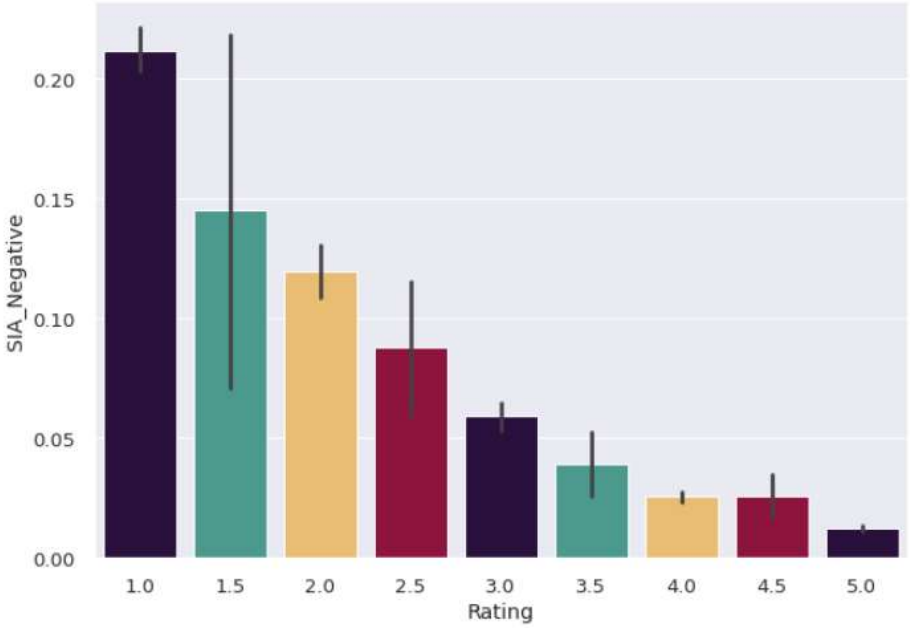
Sentiment Analysis

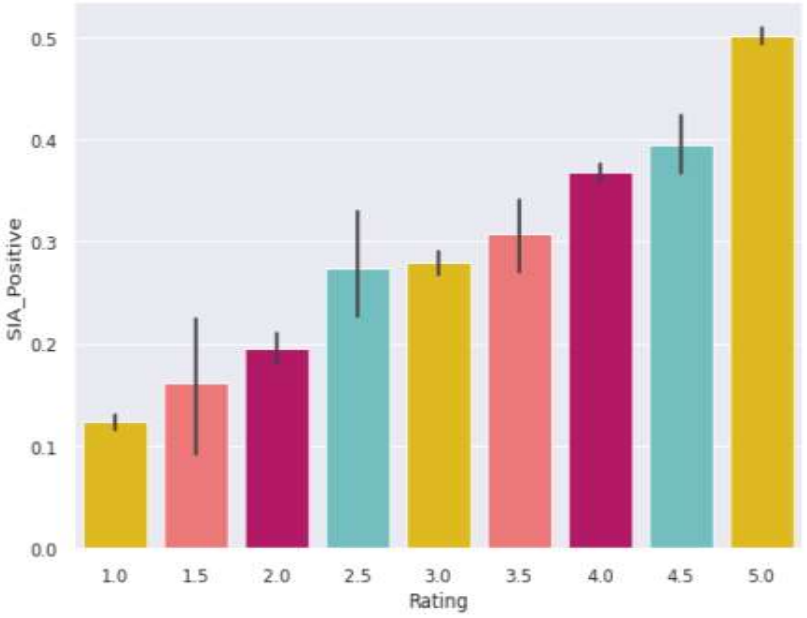
Vader Model

Sentiment analysis is a text analysis method that detects polarity (e.g. a positive or negative opinion) within the text, whether a whole document, paragraph, sentence, or clause. Sentiment analysis aims to measure the attitude, sentiments, evaluations, attitudes, and emotions of a speaker/writer based on the computational treatment of subjectivity in a text.

Sentiment Analysis (continued)

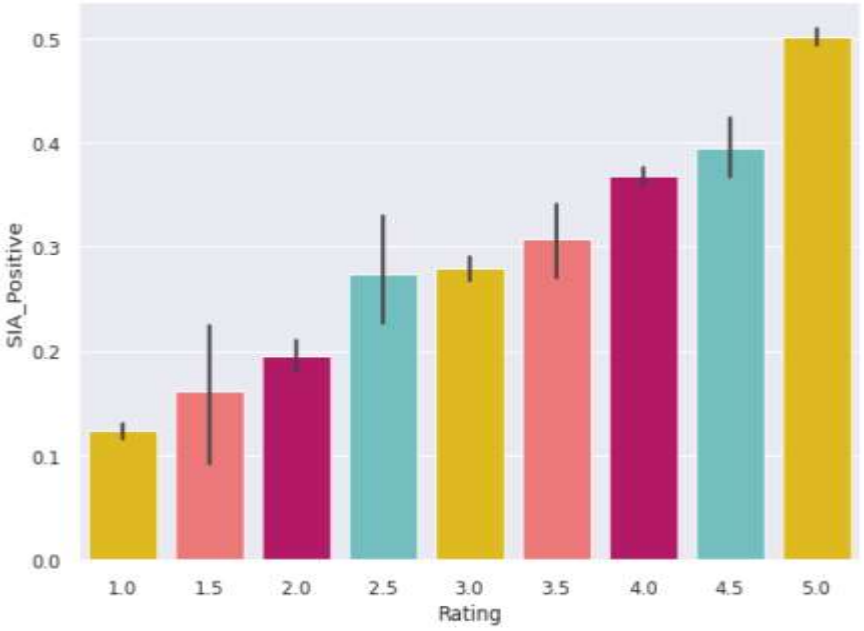
Vader Model

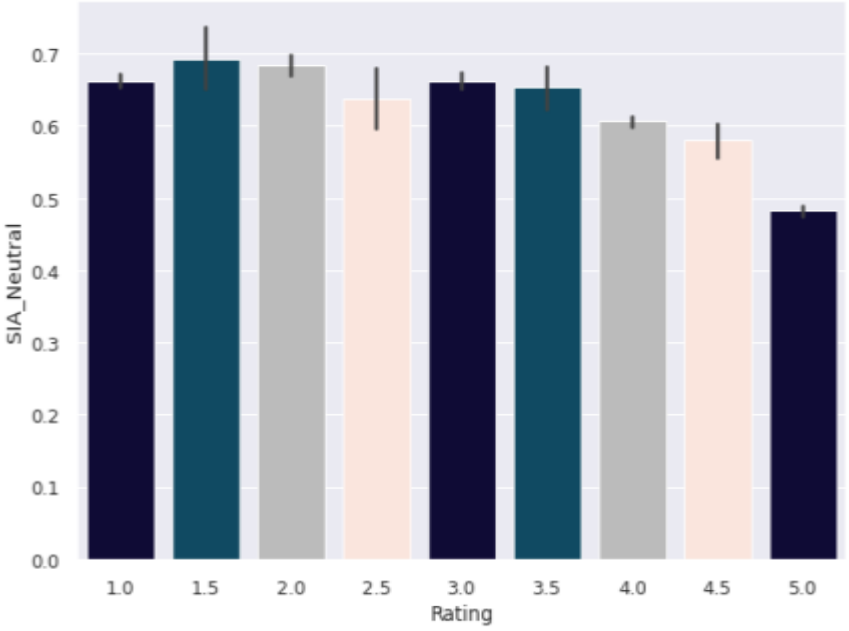




Sentiment Analysis (continued)

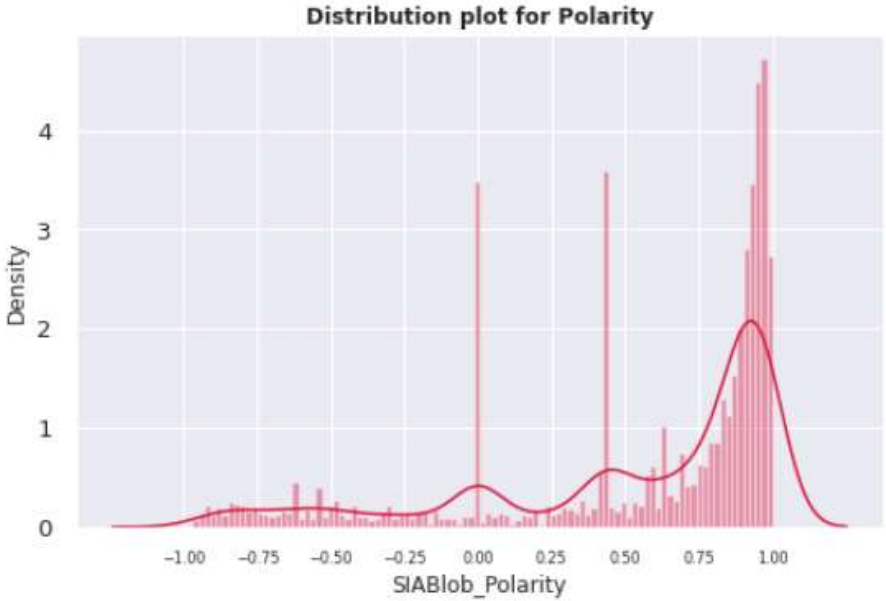
Vader Model

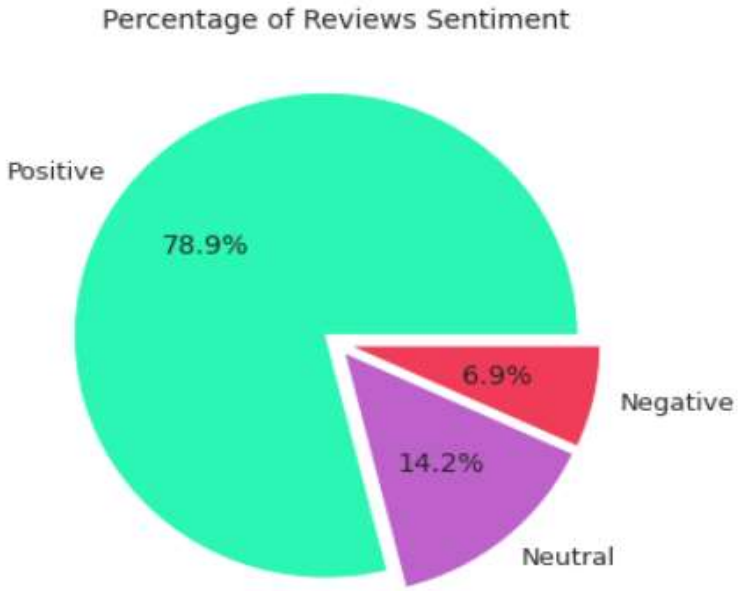




Sentiment Analysis (continued)

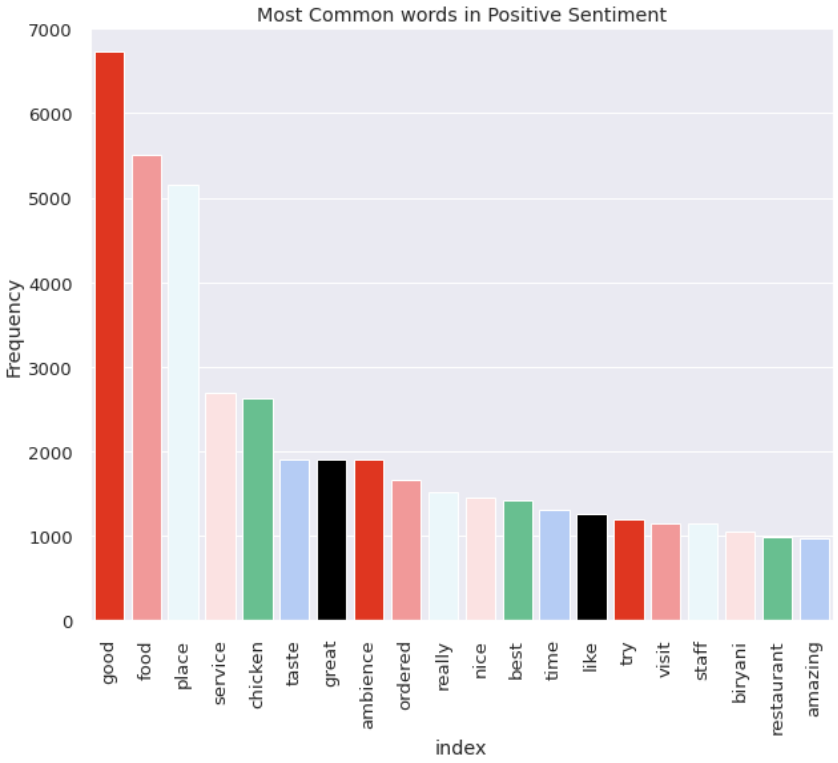
Vader Model

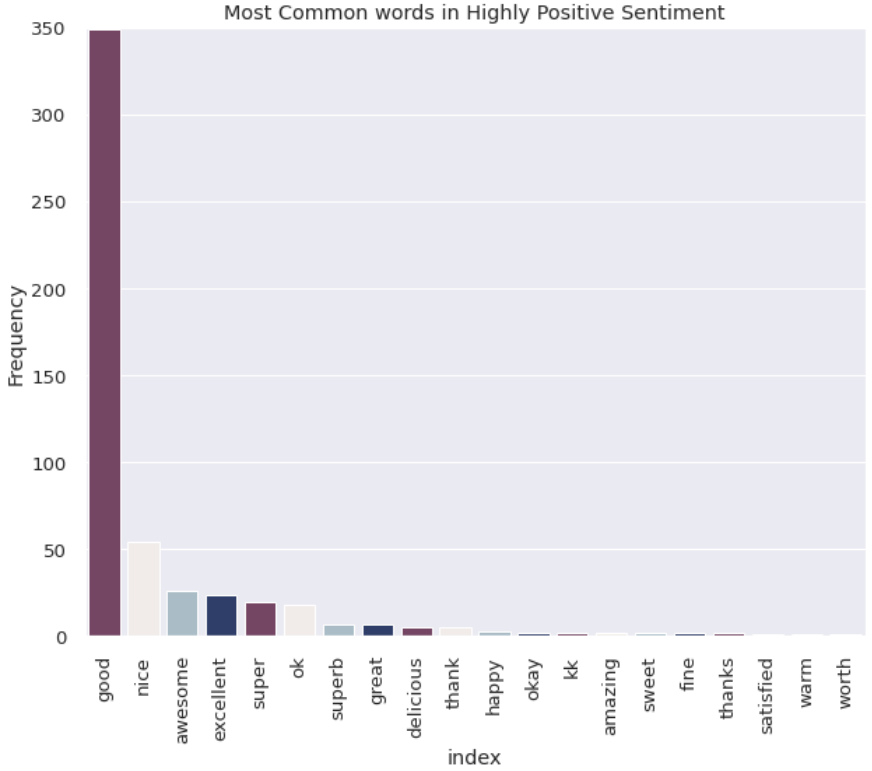




Sentiment Analysis (continued)

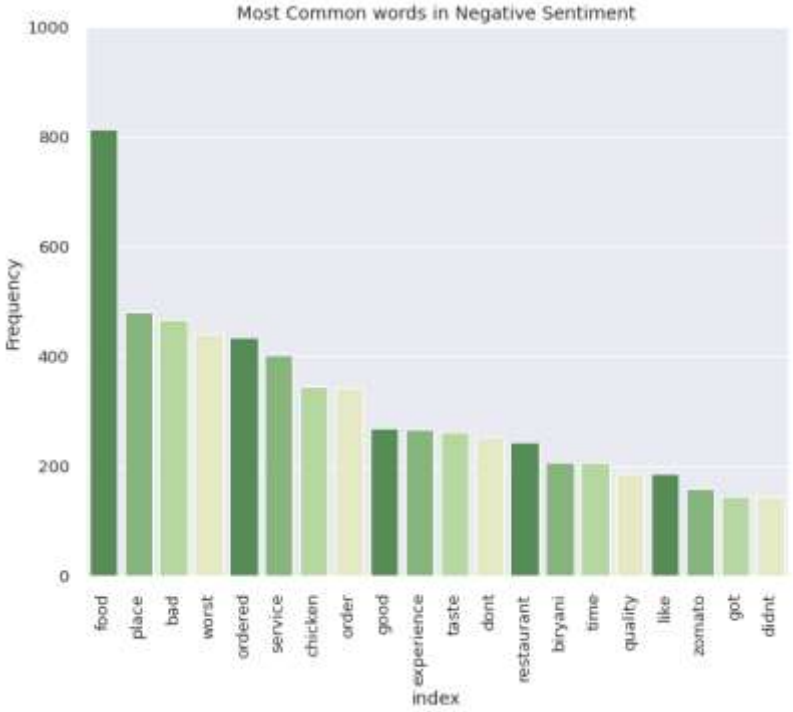
Vader Model

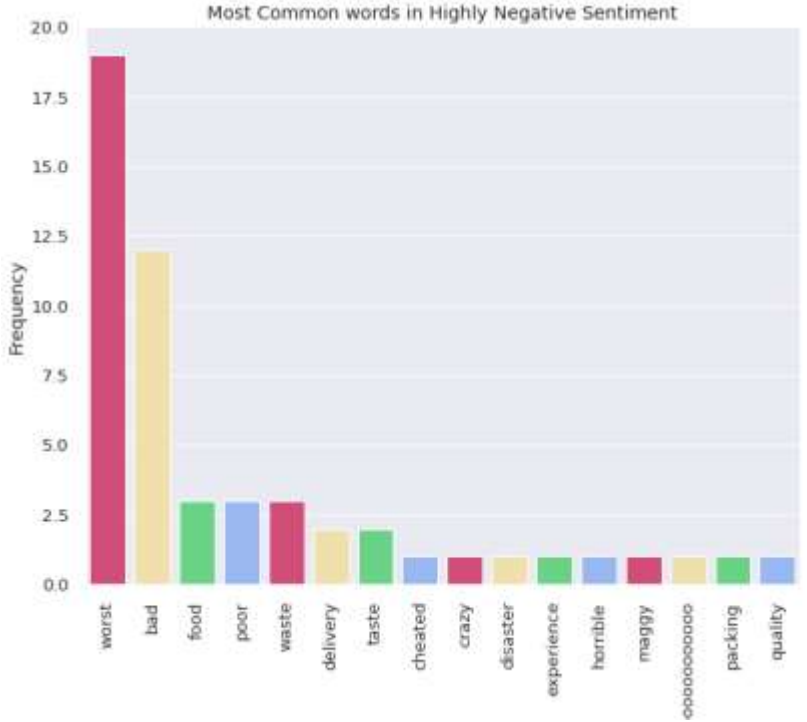




Sentiment Analysis (continued)

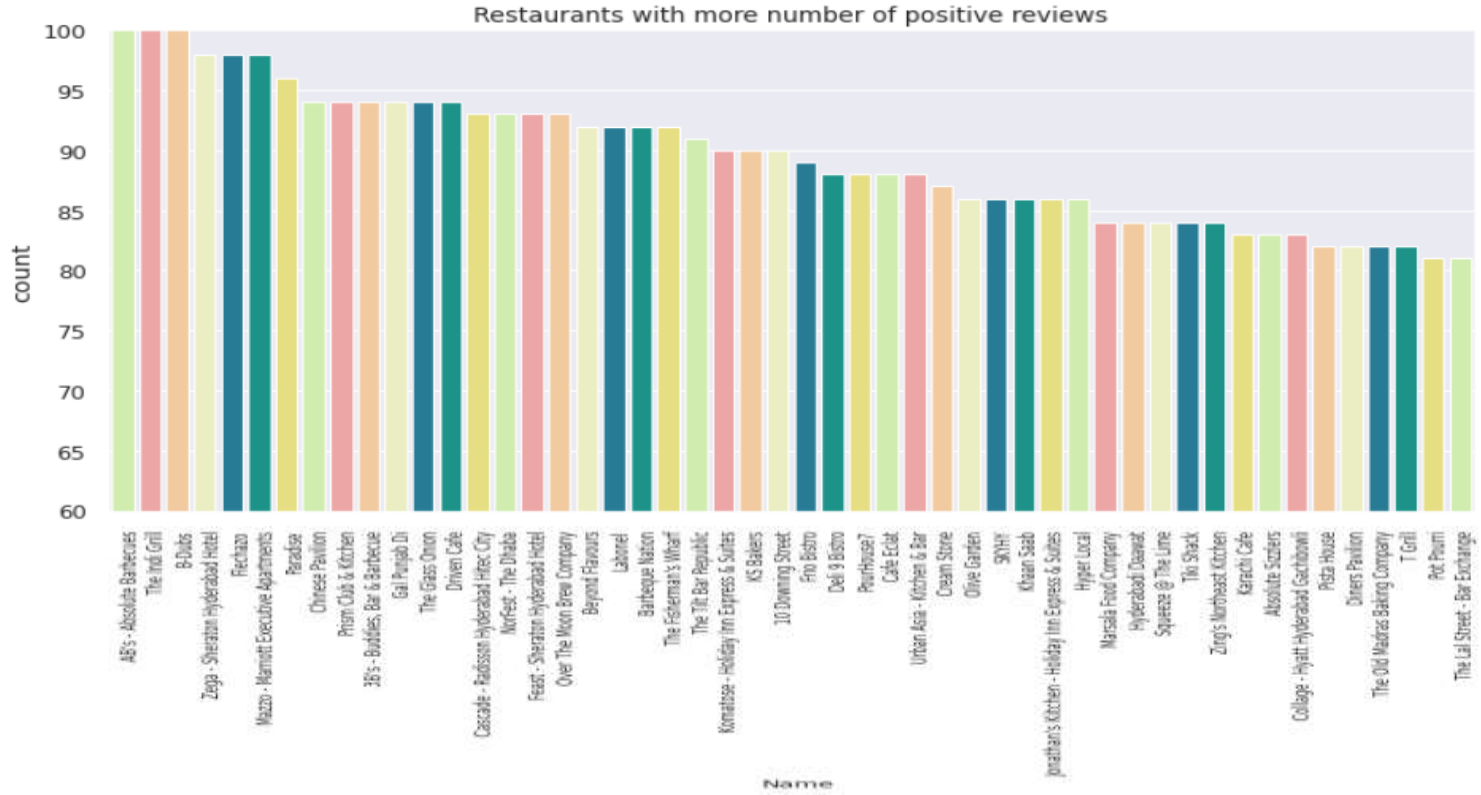
Vader Model





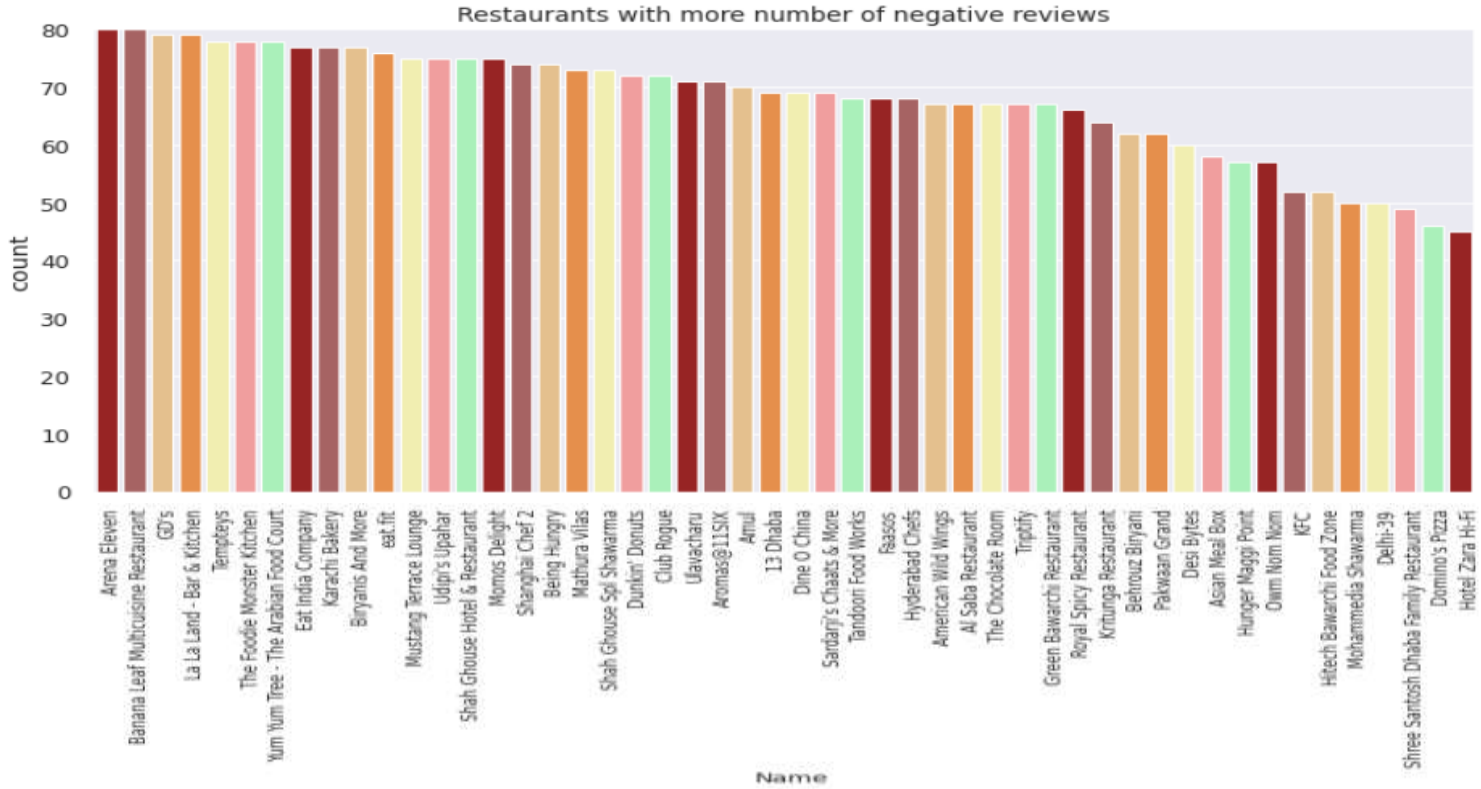
Sentiment Analysis (continued)

Vader Model



Sentiment Analysis (continued)

Vader Model



Conclusion

- **North Indian** cuisine is most common cuisine found in the restaurants.
- **Collage - Hyatt Hyderabad Gachibowli** is most expensive restaurant.
- **Amul** and **Mohammedia Shawarma** are the most affordable restaurants.
- The Restaurants are clustered on cuisines into **15** clusters by using **KMeans** clustering algorithm with the **Silhouette score** of **0.195.**
- **DBSCAN** algorithm is also used to cluster the restaurants into **15** clusters and also helps us to detect the outliers with the **Silhouette score** of **0.107.**
- **Anvesh Chowdary** has given the most number of reviews.
- **AB's – Absolute Barbecues** is the top rated restaurant.
- Almost **79** percent of the observations have **Positive** sentiment and **14** and **7** percent of the observations have **Neutral** and **Negative** sentiments respectively

Conclusion

- **Good** is the most common word in the Highly positive sentiment.
- **Worst** is the most common word in the Highly negative sentiment.
- **AB's – Absolute Barbecues, The Indi grill** and **B-Dubs** are the restaurants with the most number of positive reviews.
- **Arena Eleven** and **Banana leaf Multicuisine** restaurant are the restaurants with the most number of negative reviews.
- **Udipi's Upahar** is the most **affordable** restaurant with the **best** rating.
- **Feast - Sheraton Hyderabad Hotel** is the most **expensive** restaurant with the **best** rating.
- **Asian Meal Box** is the most **affordable** restaurant with the **worst** rating.
- **Club Rogue** is the **expensive** restaurant with **worst** rating.

Thank you