Zomato Restaurant Clustering & Sentiment Analysis

Sandhya kumari sah
Data science trainees,
AlmaBetter, Bangalore

Introduction:

Zomato is an Indian multinational restaurant aggregator and food delivery company founded by Deepinder Goyal, Pankaj Chaddah, and Gunjan Patidar in 2008. Zomato provides information, menus, and user reviews of restaurants as well as food delivery options from partner restaurants in select cities. Zomato had also made a name for itself for its prowess in digital marketing.

Like most other startups, India's pioneering food tech unicorn Zomato has seen many peaks and troughs in its journey. While there were some illustrious moments and accomplishments, there were troubled times too, some that even brought the very existence of the company into question. While Zomato competes with Swiggy, UberEats, and Ola Foodpanda among others, the company continues to innovate offerings and expand in the manner it has been doing since the day it all started.

On the other hand, Hyderabad, capital and largest city of Telangana, one of the most ethnically diverse cities in the country, with over 51% of the city's population being migrants from other parts of India. Hyderabad has a unique food culture. Restaurants from all over the world can be found here, with various kinds of cuisines.

Food apps like Zomato provide a secular part where users can rate their experience of the visited restaurant or café. Zomato also provides columns for writing classified user reviews. Sharing on the internet is something we usually do. Giving a review is also a useful activity so that other people on the internet can find out something else and see opinions about things. Food apps like

Zomato provide a secular part where users can rate their experience of the visited restaurant or café. Zomato also provides columns for writing classified user reviews. Sharing on the internet is something we usually do. Giving a review is also a useful activity so that other people on the internet can find out something else and see opinions about things.

So, in this article, we will be analyzing the Zomato restaurant data to identify the unique food culture and cuisines of Hyderabad, we will try to solve business cases that can directly help the

customers find the best affordable restaurant in their locality. We will be grouping restaurant data on the basis of cost, ratings and cuisines served, will perform topic modelling on reviews and perform sentiment analysis on review data.

Problem Statement:

Zomato is an Indian restaurant aggregator and food delivery start-up founded by Deepinder Goyal and Pankaj Chaddah in 2008. Zomato provides information, menus and user-reviews of restaurants, and also has food delivery options from partner restaurants in select cities.

India is quite famous for its diverse multi cuisine available in a large number of restaurants and hotel resorts, which is reminiscent of unity in diversity. Restaurant business in India is always evolving. More Indians are warming up to the idea of eating restaurant food whether by dining outside or getting food delivered. The growing number of restaurants in every state of India has been a motivation to inspect the data to get some insights, interesting facts and figures about the Indian food industry in each city. So, this project focuses on analyzing the Zomato restaurant data for each city in India.

The Project focuses on Customers and Company, we will analyze the sentiments of the reviews given by the customer in the data and made some useful conclusion in the form of Visualizations. Also, cluster 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.

This could help in clustering the restaurants into segments. Also, the data has valuable information around cuisine and costing which can be used in cost vs. benefit analysis

Data could be used for sentiment analysis. Also, the metadata of reviewers can be used for identifying the critics in the industry.

Dataset Analysis:

We will combine two datasets for this project.

Zomato Restaurant names and Metadata: This dataset contains restaurant related info.

1. Name: Name of Restaurants
2. Links: URL Links of Restaurants
3. Cost: Per person estimated Cost of dining
4. Collection: Tagging of Restaurants w.r.t. Zomato categories
5. Cuisines: Cuisines served by Restaurants
6. Timings: Restaurant Timings

Zomato Restaurant reviews: This dataset contains critic info and reviews.

1. Restaurant: Name of the Restaurant
2. Reviewer: Name of the Reviewer
3. Review: Review Text
4. Rating: Rating Provided by Reviewer
5. MetaData: Reviewer Metadata - No. of Reviews and followers
6. Time: Date and Time of Review
7. Pictures: No. of pictures posted with review

Exploratory Data Analysis:

Exploratory Data Analysis or EDA perform a key role to become acquainted with data to drive intuition and begin to formulate testable hypothesis. This process typically makes use of descriptive statistics and visualizations.

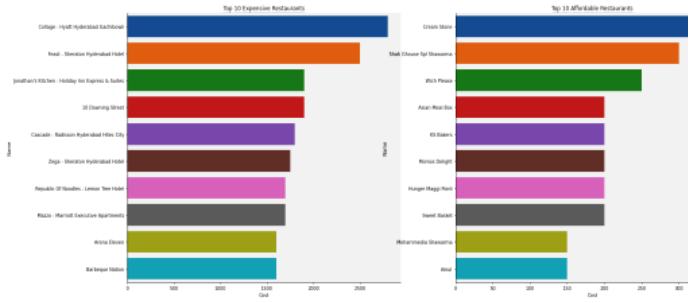


Fig 1. Top 10 most Expensive and Affordable Restaurants

Peoples prefers to give ratings in full number. Most of the ratings are 5 or 4 starred

Fig 4. Avg Ratings vs Cost

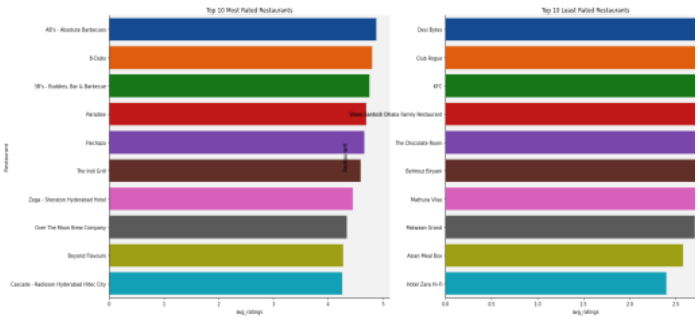


Fig 2. Top 10 most Rated and least Rated Restaurants

Though a clear linear relationship cannot be observed, but normally we can see some dense grouped low budget restaurants with low avg ratings.

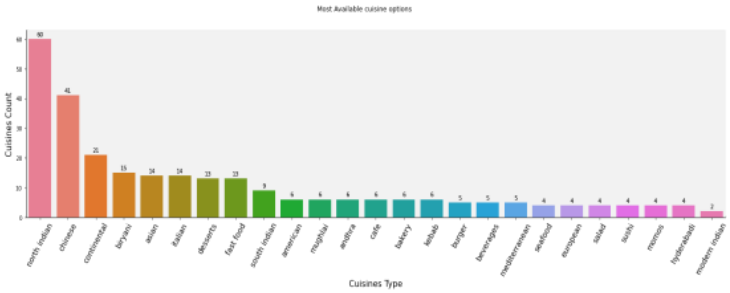


Fig 5. Chances of CHD over ages

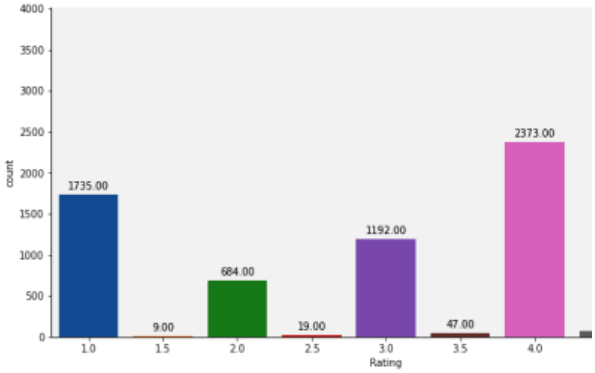


Fig 3. Distributions of Ratings

Fig 7. Ratings of the Restaurants by the Top Reviewers

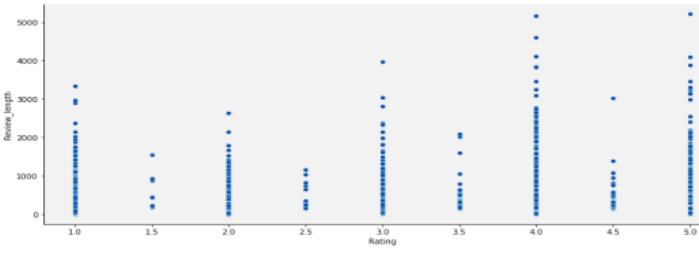


Fig 9. Rating vs Review Length

Fig 8. Wordcloud of Most occurred words in Positive & Negative Reviews

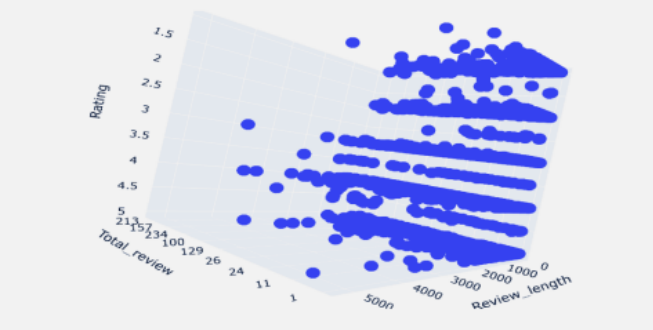


Fig 10. Review Length vs Rating vs Number of Reviews

Clustering:

As we've 42 different cuisine options available, we've grouped these into five parent categories, or else data will be of much higher dimensional results in poor clustering. Along with these we've considered Cost and avg rating for clustering too.

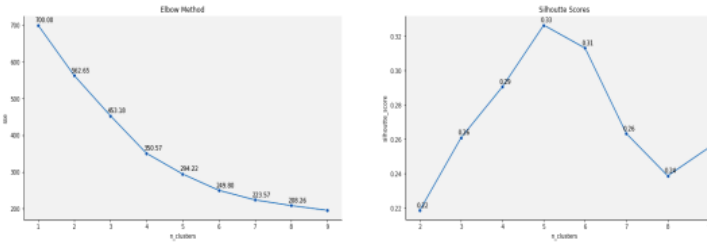
The cuisine option we grouped like below,

- ['continental', 'european', 'bbq', 'mexican', 'spanish', 'wraps', 'american', 'italian', 'fast food', 'mediterranean', 'burger', 'european', 'pizza'] → Continental
- ['thai', 'asian', 'japanese', 'chinese', 'indonesian', 'momos', 'seafood', 'sushi', 'healthy food'] → Oriental
- ['bakery', 'ice cream', 'finger food', 'juices', 'desserts', 'cafe', 'beverages', 'salad'] → Dessert
- ['hyderabadi', 'north indian', 'modern indian', 'street food', 'south indian', 'biryani', 'kebab', 'andhra', 'mughlai', 'hyderabadi', 'goan', 'north eastern'] → Indian
- [Any other except these] → Others

We've used StandardScaler from sklearn to normalize our data before clustering. 3 types of clustering algorithms have been used.

1. **Kmeans**: K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning or data science. It groups the unlabeled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on.

Fig 11. Elbow Method & Silhouette Score Line plot



From the elbow method, we're getting our elbow on n_clusters = 4, but from the silhouette score, it is max for n_clusters = 5. We know, elbow method is absolutely based on one's perception, so we will plot silhouette score for clusters range of 3 to 6.

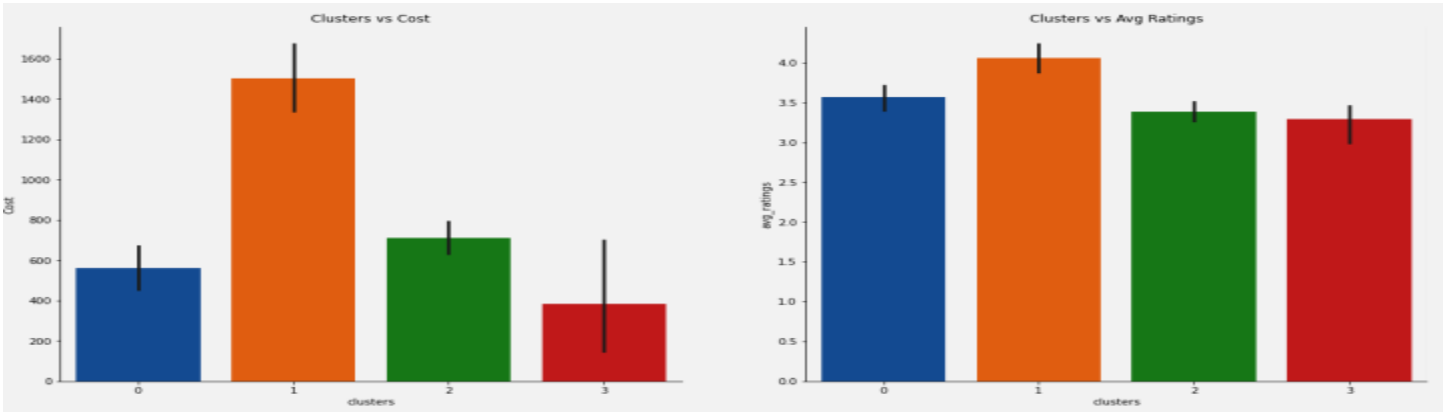


Fig 13. Cost and Avg rating against Clusters

We will use t-SNE to display clusters against cuisines type. t-SNE helps make the cluster more accurate because it performs dimensionality reduction and converts data into a 2d or 3d space where dots are in a circular shape (which pleases to k-means and it's one of its weak points when creating segments.

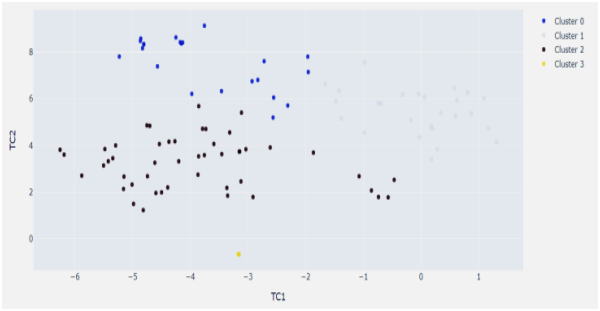


Fig 14. Visualizing clusters in 2d using t-SNE (perplexity = 40)

Cluster 3 is separated and has very less amount of data within it. Cluster 0 has some data plotted very close to other clusters. Cluster 1 and Cluster 2 can be clearly differentiated.

We can notice the clusters were not very clearly separated.

2. **Agglomerative Clustering**: The agglomerative clustering is the most common type of hierarchical clustering used to group objects in clusters based on their similarity. It's also known as AGNES (Agglomerative Nesting). The algorithm starts by treating each object as a singleton cluster. Next, pairs of clusters are successively merged until all clusters have been merged into one big cluster containing all objects. The result is a tree-based representation of the objects, named dendrogram.

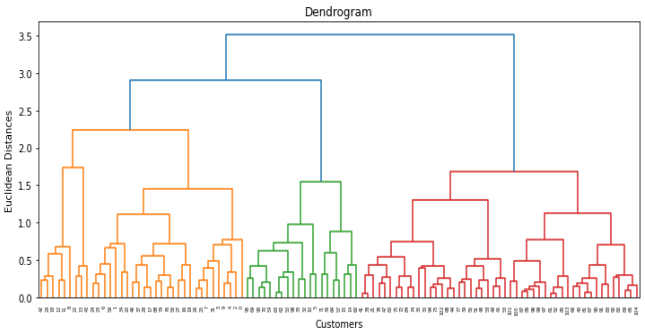


Fig 16. Dendrogram with Horizontal Threshold Line

Looking at the dendrogram, we've plotted a horizontal line up to which no vertical line intersects with any clusters. Given that 6 vertical lines cross the threshold, the optimal number of clusters is 6.

3. **DBScan**: DBSCAN stands for Density-Based Spatial Clustering. DBSCAN clustering is an underrated yet super useful clustering algorithm for unsupervised learning problems. 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. It also does not require the number of clusters to be told beforehand, unlike K-Means, where we have to specify the number of centroids.

   DBSCAN requires only two parameters: epsilon and minPoints. Epsilon is the radius of the circle to be created around each data point to check the density and minPoints is the minimum number of data points required inside that circle for that data point to be classified as a Core point.

Normally, in case of dimension greater than 2, min samples should be 2*Dimensions.

We will use K-Nearest Neighbor to obtain epsilon using maximum curvature of graph technique.

By looking at the curve visually, it looks like maximum curvature of the curve

is about 1.75. So, we took epsilon as 1.75 and min points as 2*7 = 14 (as we've 7 features within).

Noisy grouped restaurants have higher avg cost, it's because those are expensive and exclusive, so low in count also doesn't match with others common kitchens.

We can notice two clearer groups in this case with some noise though.

Unsupervised Sentiment Analysis (Topic Modelling):

Topic Modelling is different from rule-based text mining approaches that use regular expressions or dictionary based keyword searching techniques. It is an unsupervised approach used for finding and observing the bunch of words (called "topics") in large clusters of texts.

Topics can be defined as "a repeating pattern of co-occurring terms in a corpus". A good topic model should result in – "health", "doctor", "patient", "hospital" for a topic – Healthcare, and "farm", "crops", "wheat" for a topic – "Farming".

Topic Models are very useful for the purpose for document clustering, organizing large blocks of textual data, information retrieval from unstructured text and feature selection. For Example – New York Times are using topic models to boost their user – article recommendation engines. Various professionals are using topic models for recruitment industries where they aim to extract latent features of job descriptions and map them to right candidates. They are being used to organize large datasets of emails, customer reviews, and user social media profiles.

• Text Preprocessing:

Text preprocessing is traditionally an important step for natural language processing (NLP) tasks. It transforms text into a more digestible form so that machine learning algorithms can perform better.

1. Lower Case Texts: This process converts all letters to its lowercase format. This helps to reduce dimension and it is a great way to deal with sparsity issues.
2. Tokenization: This process split the sequence of strings into words. It removes all the punctuations from the text data and gives words of text which is called tokens. Basically, it split the words on the basis of non-letter characters like space, commas, full stop or non utf-8 compliant whitespaces. In this step, all punctuation marks and whitespaces are removed.
3. Regular Expression Based Processing: Any of pasted or attached link has also been replaced with single common word "url". Also, any 3 consecutive letters have been replaced with 2 letters of the same alphabet.
4. Filter Tokens by Length: This is to remove all irrelevant small length tokens which have no greater importance in our model. Any tokens having a length less than 3 has been removed from corpus. 5. Remove Stopwords: This process removes words from the document which does not play any important in giving intelligent pattern or information. Ex: the words like "as", "by", "are", "then", "with" etc.

After applying all above-mentioned preprocessing steps, a sample text would like below,

Sample Text:

Post Processed Text:

• Text Normalization:

Text Normalization is a process that converts a list of words to a more uniform

sequence. This is useful in preparing text for later processing and also it does improve text matching. Stemming and Lemmatization are two techniques used for text normalization. We've used the lemmatization to normalize tokens before vectorize it.

Lemmatization is a linguistic term that means grouping together words with the same root or lemma but with different inflections or derivatives of meaning so they can be analyzed as one item. The aim is to take away inflectional suffixes and prefixes to bring out the word's dictionary form.

For example, to lemmatize the words "cats," "cat's," and "cats'" means taking away the suffixes "s," "'s," and "s'" to bring out the root word "cat." Lemmatization is used to train robots to speak and converse, making it important in the field of artificial intelligence (AI) known as "natural language processing (NLP)" or "natural language understanding."

One widely known application of lemmatization is information retrieval for search engines. Lemmatization allows systems to map documents to topics, allowing search engines to display relevant results and even expanding them to include other information that readers may find useful, too.

Lemmatization is also used in sentiment analysis, which includes text preparation before examination. The concept is also applied in document clustering, where users need to extract topics and retrieve information.

• LDA (Latent Dirichlet Allocation):

LDA is a popular topic modeling technique to extract topics from a given corpus. The term latent conveys something that exists but is not yet developed. In other words, latent means hidden or concealed. Now, the topics that we want to extract from the data are also "hidden topics". It is yet to be discovered. Hence, the term "latent" in LDA. The Dirichlet allocation is after the Dirichlet distribution and process.

The LDA makes two key assumptions:

1. Documents are a mixture of topics, and
2. Topics are a mixture of tokens (or words)

We've used LDA with Bag of Words (Count Vectorizer) with hyper parameter tuning done on LDA model.

Before hyper parameter tuning, the acquired metrics are,

1. Log Likelihood Score: -1984697.3822986437 (the more the better)
2. Model Perplexity: 1861.9365422499502 (the less the better)

After hyper parameter tuned, the acquired metrics are,

1. Log Likelihood Score: -452717.07873869175 (the more the better)
2. Model Perplexity: 1797.1764277422712 (the less the better)

Best model parameter we've achieved are,

1. learning_decay: 0.5
2. n_components: 3

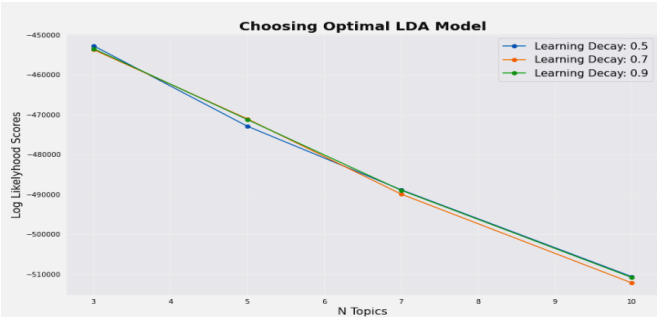
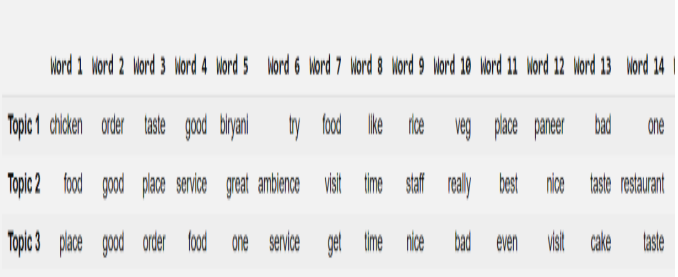


Fig 24. Log Likelihood Scores vs N number of Topics

We get 3 topics as our desired one. The top 15 words in every topics are as below,



	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8	Word 9	Word 10	Word 11	Word 12	Word 13	Word 14
Topic 1	chicken	order	taste	good	biryani	try	food	like	rice	veg	place	paneer	bad	one
Topic 2	food	good	place	service	great	ambience	visit	time	staff	really	best	nice	taste	restaurant
Topic 3	place	good	order	food	one	service	get	time	nice	bad	even	visit	cake	taste

Fig 25. Top 15 words in every topic

Supervised Sentiment Analysis:

Topic Modelling is different from rule NLP or Natural Language Processing is a new emerging hot topic in field of Data Science & Machine Learning. NLP is used to interpret human language and behavior.

Sentiment Analysis is one of the convenient applications of NLP. It does the task of classifying the polarity of a given text. For instance, a text-based tweet can be categorized into either "positive", "negative", or even "neutral" also. Given the text and accompanying labels, a model can be trained to predict the correct sentiment. Vader, TextBlob, Google Cloud Natural Language API are examples of some of the pretrained model for sentiment analysis.

We've already prepared tokenized clean texts. As our data doesn't have labels associated with, so all the reviews with greater than or equal to 3.5 rating are labelled as "Positive (1)", else every were labelled as "Negative (0)". The clean texts were then passed through a pipeline. A pipeline is a linear sequence of data preparation options, modeling operations, and prediction transform operations. It allows the sequence of steps to be specified and evaluated. As an atomic unit, the pipeline can be evaluated using a preferred resampling scheme such as a train-test split or k-fold cross validation. Pipeline is beneficial to prevent data leakage, and to add reproducibility.

Initially, we've split our data to 4:1 ratio, this would separate a set of records to validate on pipelines. Four different pipelines were created, all of which has the predefined first step of text vectorization using BOW. To implement BOW technique, CountVectorizer was used. Those pipelines were trained, we will discuss about all the metrics obtained below, before that the last step of all pipelines was, classification with ML Algorithms discussed earlier. Once the all the steps were defined, the data are split using ShuffleSplit, which will randomly sample entire training dataset during each iteration to divide it further into train and test, which are to be used for cross validation using HalvingGridSearch supported by scikit-learn. Halving grid search over specified parameter values with successive halving. The search strategy starts evaluating all the candidates with a small number of resources and iteratively selects the best candidates, using more and more resources, thus results in most similar result with impressive lesser time than its closest GridSearchCV.

The following metrics were used to evaluate the performance of our pipelines.

- **Accuracy:** Accuracy is the quintessential classification metric. Accuracy is the proportion of true results among the total number of cases examined. It is easily suited for binary as well as a multiclass classification problem. Accuracy is a valid choice of evaluation for classification problems which are well balanced and not skewed or no/less class imbalance.

  Accuracy = (TP + TN) / (TP + TN + FP + FN)

❖ **Precision (Macro):** Precision answers what proportion of predicted positives is truly positives. Precision is a valid choice of evaluation metric when we want to be very sure of our prediction.

Precision Macro = (Sum of Precision for each individual class) / (No. of Classes)

❖ **Recall (Macro):** Recall answers what proportion of actual Positives is correctly classified. Recall is a valid choice of evaluation metric when we want to capture as many positives as possible.

Recall Macro = (Sum of Recall for each individual class) / (No. of Classes)

❖ **F1 Score:** The F1 score is a number between 0 and 1 and is the harmonic mean of precision and recall. F1 score sort of maintains a balance between the precision and recall for classifier. If precision is low, the F1 is low and if the recall is low again F1 score is low. The F1 score manages the tradeoff.

F1 Score = 2 * (Precision * Recall) / (Precision + Recall)

❖ **AUC:** AUC is the area under the ROC curve. AUC ROC indicates how well the probabilities from the positive classes are separated from the negative classes. AUC is scale-invariant. It measures how well predictions are ranked, rather than their absolute values.

Five different models were trained with our pipeline.

➢ **BernoulliNB**: Bernoulli Naive Bayes is a part of the family of Naive Bayes. It only takes binary values. The most general example is where we check if each value will be whether or not a word that appears in a document. That is a very simplified model. In cases where counting the word frequency is less important, Bernoulli may give better results. In simple words, we have to count every value binary term occurrence features i.e., a word occurs in a document or not. These features are used rather than finding the frequency of a word in the document.

➢ **LinearSVC**: Linear Support Vector Classification. Similar to SVC with parameter kernel='linear', but implemented in terms of liblinear rather than libsvm, so it has more flexibility in the choice of penalties and loss functions and should scale better to large numbers of samples. Specifies the loss function. Linear Kernel is commonly recommended for text classification because most of these types of classification problems are linearly separable. The linear kernel works really well when there are a lot of features, and text classification problems have a lot of features. Linear kernel functions are faster than most of the others and you have fewer parameters to optimize. The LinearSVC module from sklearn library provides support for linear support vector machine. The function for linear kernel is as below,

$$f(X) = wT * X + b$$

➢ **Logistic Regression**: Logistic

regression is a statistical analysis method to predict a binary outcome, such as yes or no, based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables. For example, a logistic regression could be used to predict whether a political candidate will win or lose an election or whether a high school student will be admitted or not to a particular college. These binary outcomes allow straightforward decisions between two alternatives. Logistic regression can also estimate the probabilities of events, including determining a relationship between features and the probabilities of outcomes. It assumes that,

- The dependent variable is binary or dichotomous—i.e. It fits into one of two clear-cut categories. • There should be no, or very little, multicollinearity between the predictor variables—in other words, the predictor variables (or the independent variables) should be independent of each other.

➢ **Decision Tree Classifier**: A decision tree is a tree-like graph with nodes representing the place where we pick an attribute and ask a question; edges represent the answers the to the question; and the leaves represent the actual output or class label. They are used in non-linear decision making with simple linear decision surface. Decision trees classify the examples by sorting them down the tree from the root to some leaf node, with the leaf node providing the classification to the example. Each node in the tree acts as a test case for some attribute, and each edge descending from that node corresponds to one of the possible answers to the test case. This process is recursive in nature and is repeated for every subtree rooted at the new nodes. A general algorithm for a decision tree can be described as follows:

- Pick the best attribute/feature. The best attribute is one which best splits or separates the data.
- Ask the relevant question.
- Follow the answer path.
- Go to step 1 until you arrive to the answer.

➢ **K-Neighbors Classifier**: The k-nearest neighbors' algorithm, also known as KNN or k-NN, is a non parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another. For classification problems, a class label is assigned on the basis of a majority vote—i.e., the label that is most frequently represented around a given data point is used.

Below is the list of models' performance ascending

by their performance on the scale of f1 score.

Model Selection (Logistic Regression):

Final model has been built using Logistic Regression, with hyper parameter tuning done on both bag of words and ml model. The parameters are,

1. CountVectorizer: max_df = 1.0; max_features = 10000

2. LinearSVC: penalty = l2; class_weight = balanced; max_iter = 100000; C = 0.1

Confusion Matrix:

Balanced Accuracy Score: 0.8449

Average Weighted F1 Score: 0.8508

ROC AUC Graph:

Limitation:

- Finding optimum number of clusters is purely a visual job and depends on one who analyzing. It's very challenging and is keen prone to human error.
- Our data contains very a smaller number of restaurants, as well as user reviews also contains a very limited set of vocabulary. These affects performing topic modelling and sentiment analysis too.

Scope of Improvement:

- We've grouped cuisines by demographical origin and availability. We can try to group cuisine by dish serve type or food type category. This manipulates the clusters most.
- We can try some other vectorization method like Inverse Document Term Frequency etc. Some advanced complex word embedding techniques can also be applied like Glove, doc2vec but those work greatly on larger document corpus, we don't think with such limited data we've those complex methods could perform any better.

Conclusion:

- Best optimum number of clusters we obtained is 4.
- We've achieved 3 distinct topics as best component no for Topic Modelling.
- For Sentiment Analysis, Logistic Regression is the highest accurate model so far. Avg Weighted F1 score is 0.85 and the area under roc curve is 0.91 which is satisfactory.
- Interesting fact, longer reviews have more chance of having a higher rating.

References:

➢ https://towardsdatascience.com/how-to-create-dynamic-3d-scatter-plots-with-plotly-6371adafd14

➢ https://www.datatechnotes.com/2020/11/tsne-visualization-example-in python.htmlhttps://www.datatechnotes.com/2020/11/tsne-visualization-example-in-python.html

➢ http://www.sefidian.com/2020/12/18/how-to-determine-epsilon-and-minpts-parameters-of-

dbscan-clustering/ ➢
https://highdemandskills.com/topic-model-evaluation/

THE END