Capstone Project

Zomato Restaurant Clustering and Sentiment Analysis

Technical Documentation

By

Girish R

Data Science Trainee, Alma Better

Bangalore

Abstract:

In today's digital world, food apps like Zomato are widely used because it provides a platform for people to share their opinion about the restaurants and cafes they have visited. This paper includes an analysis of client ratings and reviews in Zomato utilizing content mining. Utilizing content mining, break down the content audits/reviews from the client with a specific end goal to create productive results and legit surveys. The rating has a review of the restaurant which can be used for sentiment analysis. Based on this, writers want to discuss the sentiment of the review to be predicted. The method used for preprocessing the review is to make all words lowercase, tokenization, remove numbers and punctuation, stop words, and lemmatization. Then after that, we create a word to vector with the term frequency-inverse document frequency (TF-IDF). The data that we process are 10,000 reviews. After that, we make positive reviews that have a rating of 3.5 and above, negative reviews that have a rating of 3 and below. We have used Split Test, 75% Data Training and 25% Data Testing. The metrics used to determine classifiers are precision, recall, accuracy, F1 score.

Problem Statement:

The Project focuses on Customers and Company, you have to analyze the sentiments of the reviews given by the customer in the data and make some useful conclusions in the form of Visualizations. Also, cluster the Zomato restaurants into different segments. The data is visualized as it becomes easy to analyze data in an instant. The Analysis also solves some of the business cases that can directly help the customers find the best restaurant in their locality and for the company to grow up and work in 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 costs 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.

Introduction:

In today's digitized modern world, the popularity of food apps is increasing due to their functionality to view, book, and order food with a few clicks on the phone for their favorite restaurant or cafes, by surveying the user ratings and reviews of the previously visited customers. 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. The usual things are reviewed by someone in the form of experiences, places, objects, and others. When giving a review we usually use text to explain something that we experience with an item, place, or event that we normally experience.

Zomato is a site where someone can give a review of a restaurant, how the restaurant is, and someone's opinion about the restaurant. Restaurant customer satisfaction can be analyzed by their review on Zomato. Sometimes, restaurants see the reviews in Zomato, but they don't get if the reviews are positive or negative to their restaurants. Reviews on Zomato are still in the form of text and can be classified with positive, negative, or neutral ratings. Zomato doesn't have an analysis of how users interact with the reviews and what words will indicate whether they like it or not. We need to extract the words in review and analyze them so we can know how users interact in Zomato and get customers' satisfaction by their reviews.

In this paper, we propose a method to analyze users' sentiment of Zomato Restaurants. We are using different classifiers to classify the sentiments of users based on their reviews. We also find words that affect the classifier model. Also, we focus on mining customer reviews, authenticating them, and classifying them into positive and negative reviews. We also clustered the restaurants based on their cuisines.

Data Summary:

Zomato Restaurant names and Metadata

- Name: Name of Restaurants
- Links: URL Links of Restaurants
- Cost: Per person estimated Cost of dining
- Collection: Tagging of Restaurants w.r.t. Zomato categories
- Cuisines: Cuisines served by Restaurants
- Timings: Restaurant Timings

Zomato Restaurant reviews

- Restaurant: Name of the Restaurant
- Reviewer: Name of the Reviewer
- Review: Review Text
- Rating: Rating Provided by Reviewer
- Metadata: Reviewer Metadata - No. of Reviews and followers
- Time: Date and Time of Review
- Pictures: No. of pictures posted with the review

Steps involved:

1. Null values Treatment

The data set had null values which out of which we replace some with the mean of the feature some by zero and dropped some observations which were almost filled with null values.

2. Outliers' treatment

Isolation Forests (IF), similar to Random Forests, are built based on decision trees. And since there are no predefined labels here, it is an unsupervised model Isolation Forests were built based on the fact that anomalies are the data points that are "few and different". In an Isolation Forest, randomly sub-sampled data is processed in a tree structure based on randomly selected features. The samples that travel deeper into the tree are less likely to be anomalies as they require more cuts to isolate them. Similarly, the samples which end up in shorter branches indicate anomalies as it was easier for the tree to separate them from other observations.

3. Exploratory Data Analysis

We performed univariate and bivariate analyses. This process helped us figure out various aspects and relationships among variables. It gave us a better idea of which feature behaves in which manner.

4. Encoding of categorical columns

We used One Hot Encoding (converting to dummy variables) to produce binary integers of 0 and 1 to encode our categorical features because categorical features that are in string format cannot be understood by the machine and needs to be converted to the numerical format.

5. Standardization of features

Our main motive through this step was to scale our data into a uniform format that would allow us to utilize the data in a better way while performing fitting and applying different algorithms to it.

The basic goal was to enforce a level of consistency or uniformity to certain practices or operations within the selected environment.

6. Fitting different models

For modeling, we tried various algorithms like:

Clustering

1. K means clustering
2. Hierarchical clustering

Sentiment analysis unsupervised

1. LDA

Sentiment analysis supervised.

1. Logistic Regression
2. XGBooster

Recommendation System

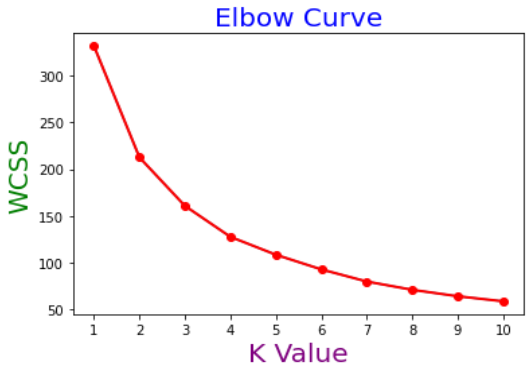
7. Tuning the hyperparameters for better recall

Tuning the hyperparameters of respective algorithms is necessary for getting better accuracy and to avoid overfitting in the case of tree-based models like Random Forest Classifier and XGBoost classifier.

Algorithms:

1. K means clustering:

K-means clustering is a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity.

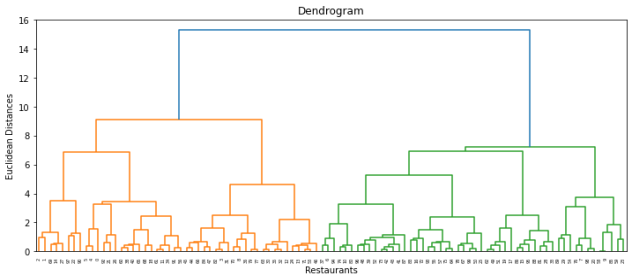


2. Hierarchical clustering:

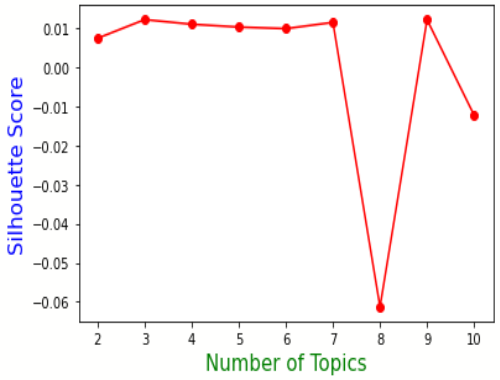
Hierarchical clustering is another unsupervised learning algorithm that is used to group together the unlabeled data points having similar characteristics. Hierarchical clustering algorithms fall into the following two categories.

Agglomerative hierarchical algorithms − In agglomerative hierarchical algorithms, each data point is treated as a single cluster and then successively merge or agglomerate (bottom-up approach) the pairs of clusters. The hierarchy of the clusters is represented as a dendrogram or tree structure.

Divisive hierarchical algorithms − On the other hand, in divisive hierarchical algorithms, all the data points are treated as one big cluster and the process of clustering involves dividing (Top-down approach) the one big cluster into various small clusters.



3. LDA:



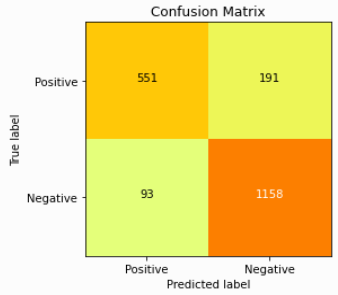
Latent Dirichlet Allocation or LDA It is one of the most popular topic modeling methods. Each document is made up of various words, and each topic also has various words belonging to it. The aim of LDA is to find topics a document belongs to, based on the words in it.

There are 2 parts in LDA:

- ✓ The ***words that belong to a document***, that we already know.
- ✓ The ***words that belong to a topic*** or the probability of words belonging into a topic, that we need to calculate.

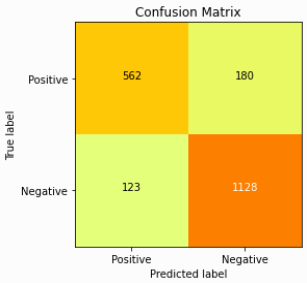
4. Sentimental Analysis (Supervised)

a. Logistic Regression



- ✓ 580 instances are labeled as True Positive (correctly predicted as positive)
- ✓ 183 instances are labeled as False Positive (incorrectly predicted as positive)
- ✓ 1148 instances are labeled as True Negative (correctly predicted as negative)
- ✓ 82 instances are labeled as False Negative (incorrectly predicted as negative)

b. XGBooster



- ✓ 445 instances are labeled as True Positive (correctly predicted as positive)
- ✓ 318 instances are labeled as False Positive (incorrectly predicted as positive)
- ✓ 1153 instances are labeled as True Negative (correctly predicted as negative)
- ✓ 77 instances are labeled as False Negative (incorrectly predicted as negative)

5. Recommendation System

Content-based filtering is a recommendation system technique that recommends items to users based on their previous preferences or interactions with items. It works by analysing the attributes of the items and user preferences, and recommending items that have similar attributes.

A user profile is created in content-based filtering, which includes information about the user's preferences, such as the types of restaurants they prefer, the types of books they read, and so on. When a user requests recommendations, the system examines the items' attributes and compares them to the user's profile. Items with attributes that correspond to the user's preferences are recommended.

In a restaurant recommendation system, for example, a user's profile may include information about the restaurant genres they prefer. If a user enjoys a particular Chinese, Italian, or Indian restaurant, the system will suggest other Chinese, Italian, or Indian restaurants to them.

Content-based filtering can also be used to suggest items to new users who have yet to interact with the system. The system will recommend items based on their attributes rather than the user's previous preferences in this case.

To improve the accuracy of recommendations, content-based filtering can be used in conjunction with other techniques such as collaborative filtering.

	USER	RESTAURANT	SCORE
73472	Shree	Hitech Bawarchi Food Zone	0.81
13143	Santosh	Karachi Cafe	0.77
23447	Kiran Thota	Al Saba Restaurant	0.87
41470	Naveen Reddy	Tiki Shack	0.73
17390	Amar	Tandoori Food Works	0.86
7241	Khaane_mey_kya_hey By Rony Samuel	Sardarji's Chaats & More	0.64
55792	Paridhi Mehra	American Wild Wings	0.63
19747	Mahesh	eat.fit	0.77
19255	Sriram Reddy	Karachi Cafe	0.80
51981	Poojitha Challagali	Shah Ghouse Spl Shawarma	0.70

Model performance:

The model can be evaluated by various metrics such as:

1. Accuracy

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. One may think that, if we have high accuracy then our model is best. Yes, accuracy is a great measure but only when you have symmetric datasets where values of false positive and false negatives are almost the same. Therefore, you have to look at other parameters to evaluate the performance of your model. For our model, we have got 0.803 which means our model is approx. 80% accurate.

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

2. Precision

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

Precision = TP/TP+FP

3. Recall

Recall (Sensitivity) - Recall is the ratio of correctly predicted positive observations to all observations in the actual class.

Recall = TP/TP+FN

4. F1-Score

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as easy to understand as accuracy, but F1 is usually more useful than accuracy, especially if you have an uneven class distribution. Accuracy works best if false positives and false negatives have similar costs. If the cost of false positives and false negatives are very different, it's better to look at both Precision and Recall.

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

Hyperparameter tuning:

Hyperparameters are sets of information that are used to control the way of learning an algorithm. Their definitions of impact parameters of the models, seen as a way of learning, change from the new hyperparameters. This set of values affects the performance, stability, and interpretation of a model. Each algorithm requires a specific hyperparameters grid that can be adjusted according to the business problem. Hyperparameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs.

We used Grid Search CV for hyperparameter tuning. This also results in cross-validation and in our case we divided the dataset into different folds.

Grid Search CV-Grid:

Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model's performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.

The Results after tunning,

- ✓ True Positive is assigned to 566 instances (correctly predicted as positive).
- ✓ False Positive is assigned to 197 instances (incorrectly predicted as positive).
- ✓ 1100 cases are classified as True Negative (correctly predicted as negative).
- ✓ False Negative is assigned to 130 instances (incorrectly predicted as negative).

Conclusion

Using a dataset of customer evaluations for the meal delivery service Zomato, clustering and sentiment analysis were conducted. To comprehend the customer's experience and learn more about their input, this analysis was conducted.

Our analysis of customer evaluations for Zomato utilized clustering and sentiment analysis to gain insights into customer satisfaction levels and identify areas for improvement. By clustering customers into positive and negative groups and using sentiment analysis to classify reviews as positive or negative, we were able to gain a comprehensive understanding of customer feedback. Our findings can be used to inform business decisions and improve the Zomato service.

From the EDA we got to know about,

- AB's - Absolute Barbecues, show maximum engagement and retention as it has maximum number of ratings on average and Hotel Zara Hi-Fi show lowest engagement as has lowest average rating.
- Most Expensive Restaurants
- Most Affordable Restaurants
- Great Buffets is the most frequently used tags.

I have chosen XGBoost model which is hyperparameter optimized for my final prediction.

- As a result of its high regularization, XGBoost is more resistant to overfitting and more adaptable to new data. With XGBoost, a supervised learning system, sentiment labels can be predicted by training on labelled data.
- The ensemble aspect of XGBoost can aid in enhancing sentiment analysis performance by pooling the predictions of various models.
- In sentiment analysis, when the model needs to generalize to new data, XGBoost's regularization can help to reduce overfitting and make the model more robust to unseen data.