Zomato Data Analysis and Restaurant Recommendation

Rahul D. Makhija
Computer Science & Engg.
PES University
Bengaluru , India
rmrahulmakhija74@gmail.com

Rishab K.S.
Computer Science & Engg.
PES University
Bengaluru , India
rishab12360@gmail.com

Rohan Mallesh
Computer Science & Engg.
PES University
Bengaluru , India
rohanmrb@gmail.com

***Abstract*—The main aim of this project is to develop an accurate model and compare different models to recommend restaurants to users. This is done by analyzing the impact of various factors that are obtained from the 'Bangalore Zomato dataset' and affect the decision of a customer. Data pre-processing and visualization is performed to understand the attributes of the dataset and make decisions for building the model. TF-IDF is used for the vectorization of text data in the attributes and models with different distance measures are built and compared with each other. Keras Embedding Layer is used to create word embeddings for text data, therefore, generating continuous numeric data for every restaurant. This numeric data of each restaurant is compared with every other restaurant and 'k' most similar restaurants are recommended.**

***Index Terms*—recommendation, data analytics, restaurant,TF-IDF**

I. Introduction

At present there is vast flow of information on the internet and it continues to grow exponentially. Despite the benefits of such humongous data, the increasing flood of information on the web creates a need for selecting content according to the end user's preferences. In this case, it is important to alter the information to a limited amount based on the current user/customer preferences in order to assist them in making the correct decision. Recommender Systems are used in a variety of areas notable for shopping, online streaming services, social media and restaurants. However users overall rating is not enough to capture the finer details for recommending any content. With the help of Deep Neural Networks Recommender Systems can now capture user interaction and a lot more detail before making any recommendation

This project aims to recommend similar restaurants to a user based on the variety of features that helps us to achieve a deep understanding of the process that makes a user choose certain restaurants over others. After the initial analysis we propose 3 different approaches for recommending restaurants on Knowledge based collaborative filtering mechanism to recommend restaurants based on user's preference, we use a standard TF-IDF vectorization function to capture data and model several recommender systems using various distance measures such as Cosine Similarity and Pearson Correlation to compare the recommendation output for a particular user input in each scenario. We also develop a Neural Network based model, where we analyze the data using word embeddings and pass it through Flatten and Dense layers to recommend restaurants based on user's preference.

II. Review of Literature

A. Multi-Criteria Review-Based Recommender System

In this paper [1], the authors surveyed a bunch of existing research papers which explore generic rating based recommender models as well as research papers on how the user-generated reviews can be utilized as an alternative and valuable source in the recommendation process through merging them with the Multi-Criteria Recommender System (MCRS) to enhance the accuracy of the Recommender System (RS)'s performance. User-generated reviews are used to improve the accuracy of the RSs performance by using text analysis and sentiment analysis to transform the unstructured user reviews into a structured form that can be merged with RSs. Hidden elements can be extracted from the user's reviews and delivering them to the RSs, tries to solve the problem of inaccurate recommendations caused by relying only on the overall ratings in the recommendation process.
The major drawback of this approach is that this approach heavily depends on the the text analysis and sentiment analysis done on users' reviews. These analytical models cannot fully comprehend or understand the user's review, therefore utilizing only part of the review.

B. A Restaurant Recommendation System by Analyzing Ratings and Aspects in Reviews

In this research paper [2], the authors have designed a restaurant recommender system based on a novel model that captures correlations between hidden aspects in reviews and numeric ratings. The authors were motivated by the observation that a user's preference against an item is affected by different aspects discussed in reviews. Their method first explores topic modeling to discover hidden aspects from review text. The authors use Latent Dirichlet Allocation (LDA) to create profiles for every user who has written a review. For predicting ratings for restaurants any user has not reviewed, Linear/Logistic Regression is employed. The authors use a browsing tool to select representative reviews for a particular restaurant.

The model was tested on 1,168,420 reviews written by 316,702 users for 42,274 restaurants in Shanghai. Since the data available is sparse, the majority of the ratings and user profiles are estimates. Hence the accuracy achieved is a mere 50 percent.

C. Restaurant Rating Prediction using Regression Models

In [3], multiple independent input attributes from the Zomato Restaurant data set are used to predict the rating of a restaurant. In this paper, the authors also do a comparative study between the performance of various regression models to arrive at a conclusion as to which one is best for this data set.
The metrics used to measure each of the regression models are regression score, absolute error, mean squared error and root mean squared error.

D. Hybrid Restaurant Recommender

In [4], the authors have presented an approach to combine the advantages of both, collaborative filtering and knowledge-based filtering. Collaborative Filtering method of mining data involves filtering of information or patterns done using techniques involving collaboration among users' viewpoints, data sources, user ratings, etc.
A hybrid model combining the advantages of both these methods is implemented. The model will choose either of the techniques based on the situation. The collaborative approach is chosen if the system knows the user well and the knowledge-based approach is chosen if the user is new to the system.
Although reported to be highly accurate, the drawback with this model is the computation cost and the cost to search in the user database. Methods such as data partitioning and multi threading are discussed to improve on the above-mentioned drawbacks.

E. Analysis of Zomato Services using Recommender System Models

In [5], the authors have proposed three different approaches which include TF-IDF based approach, Association Rule Mining Approach, and the Deep-Learning-based approach.
In the TF-IDF-based approach, which is a content-based collaborative filtering approach, they feed the cuisine type and cost as inputs to the model, and 'Cosine Similarity' is used as a metric to determine and return recommendations based on the input restaurant by the user. The second approach is based on the 'Apriori algorithm' where they recommend food items based on the frequency of the occurrences of the dish in the data set, they calculate a support vector, and association rules are formed. The third approach is the deep neural network approach where they use several of the Natural Language Processing tools like Word2Vec and FastText and pass in a custom corpus of data containing the food items and the menu to analyze the various restaurants and use cosine similarity as a metric to recommend similar restaurants.
The shortcomings of all three approaches are that all the approaches are very data-heavy and require a lot of data to be known about the restaurant to recommend a particular restaurant to a user. Additionally, too few features are considered while making a recommendation to a user in every model, thus the recommendations may not be accurate to meet the user's requirements such as the location of the restaurant. All the models fail for a new restaurant because there are very few reviews, ratings, and information about the new restaurant which makes it harder for the model to derive any knowledge about that restaurant and recommend it to a new user (Cold Start Problem).

III. Proposed Solution

A. Pre-Processing

1) Dropping columns and rows: The columns in the dataset which are not required for further analysis such as 'url', 'address' and 'phone' were dropped. After dropping these columns, presence of duplicate rows was observed and they were dropped as well.

2) Column Renaming: Few columns were not appropriately named and may mislead the exploratory data analysis. This was fixed by renaming these columns to more suitable names.

3) Removing inconsistencies: The data type of the 'rating' and 'approx cost' columns were changed to support numerical analysis. Formatting of these columns was also done.

TABLE I
Dataset Information after cleaning

Column	Count	Null/Not-Null	Dtype
name	51654	non-null	object
online_order	51654	non-null	object
book_table	51654	non-null	object
rate	41627	non-null	float64
votes	51654	non-null	int64
location	51633	non-null	object
rest_type	51427	non-null	object
dish_liked	23627	non-null	object
cuisines	51609	non-null	object
approx_cost	51309	non-null	object
reviews_list	51654	non-null	object
menu_item	51654	non-null	object
type	51654	non-null	object
neighbourhood	51654	non-null	object

B. Data Visualization

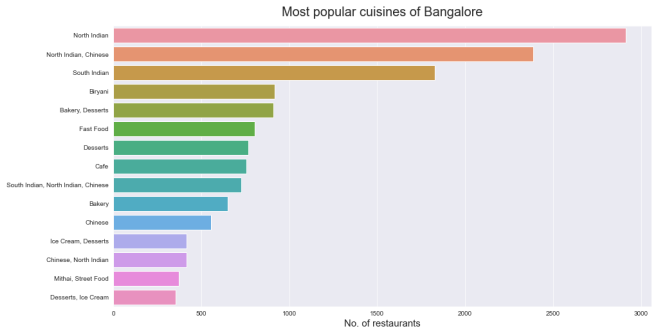

Fig. 1. Top 10 cuisine choices in Bangalore

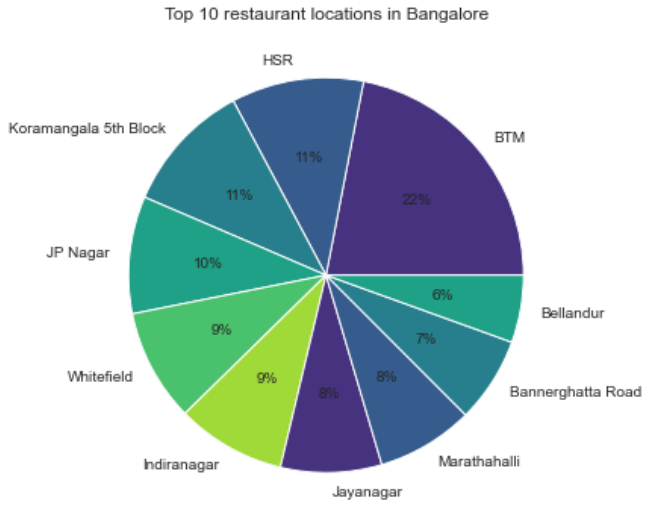

Fig. 2. Top 10 restaurant location in Bangalore

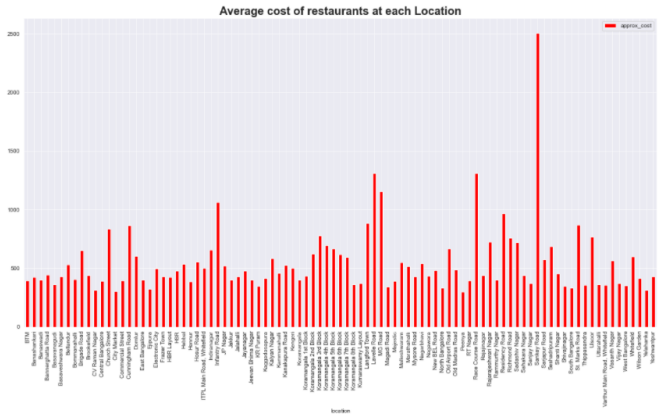

Fig. 3. Average Cost v/s Location of Restaurant

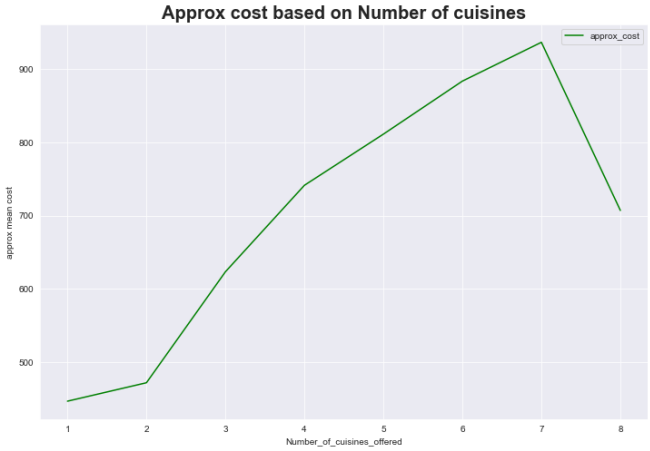

Fig. 4. Average Cost v/s Number of Cuisines

C. Model Building

1) TF-IDF - Cosine Similarity: Upon initial Data Visualization we observe a strong correlation between the rating i.e. popularity of the restaurant with the cuisine, and the location of the restaurant also upon deeper analysis we observe a Strong dependency between the avg cost of the restaurant to the location of the restaurant thus for our first approach we consider the cuisine, location, neighbourhood, approx-cost and the rating of the restaurant to recommend the users based on collaborative filtering by taking in a restaurant name as input. Initially, we combine the location and the neighbourhood columns into a single column named 'addr' to aid our comparison of the user's preferred locality.
TF-IDF is a measure of originality of the word by comparing the number of times a word occurs in a document to the number of documents the word is present. We use TF-IDF to convert our data into vector format to compare the respective features with the user's preferences. We apply TF-IDF vectorisation on the cuisine and the newly created 'location' column. After we have obtained the vectorized form of 'cuisine' and 'addr' columns, we create a N X N matrix (N = number of restaurants in the data-set) where each row represent one restaurant and we compare each restaurant with every restaurant in the data-set using the cosine similarity function by considering 'cuisine' and 'addr' features, thus yielding two N X N similarity matrices. The similarity matrix is pre-calculated and indexed using restaurant names to avoid computations for every recommendation.
The model accepts the user input and extracts the above-mentioned features with respect to the user's input from the data set. The model obtains the top 2000 similar restaurants with respect to cuisine and top 1000 similar restaurants with respect to the location from their respective matrices and makes a new list with the common restaurants from both the features. The model drops the restaurants which have an average cost difference of more than 200 units. Finally, the model drops all the duplicates in the list and sort the list according to the rating of the restaurant to output the top 10 recommendations.

2) Pearson Correlation, Euclidean Distance, Cosine Similarity: In our second approach, we build a model with various distance measures to compare the recommendations the model outputs based on various distance/similarity metrics. In this approach, we will be considering the cuisine, location, and neighbourhood columns from the data set to recommend restaurants based on user preferences. We combine all the above-mentioned features into a single column and apply TF-IDF vectorization to compare them using various distance metrics.

- Cosine Similarity: We build a N X N similarity matrix using the cosine similarity function where the model calculates similarity across all restaurants in data set, but considers only the restaurants which are more than 70 percent similar for recommendation and discards the rest.

- Euclidean Distance: The model obtains the vectorized column and applies euclidean distance measure on all the restaurants with respect to the user's preferred restaurant, the model considers only the restaurants which are less than 1 unit for the recommendation.
- Pearson Correlation: Pearson Correlation is the ratio between the covariance of two variables and the product of their standard deviations i.e. measures only linear correlation between the variables. The model obtains the Pearson correlation of all restaurants and recommends restaurants that are positively correlated and have a linear similarity of more than 70 percent.

3) Neural - Network Approach: In this approach, we are using the 'Keras Embedding Layer' to create word embeddings for every textual column (cuisine, location, dishes-liked) in the data set. The neural network consists of 3 layers - Embedding Layer, Flatten Layer and Dense Layer. The textual data is then fitted into the neural network to generate embeddings for the 3 columns for every restaurant. The above model generates a matrix of shape 8784x3 where the rows represent 8784 restaurants and the columns represent location, cuisine, dishes-liked embeddings. Using this matrix we find the 'Cosine Similarity' between every restaurant in the training data set and the user's initial choice of restaurant. We also find the Pearson's Correlation Coefficient between every restaurant in the training data set and the user's initial choice of restaurant. We set up a threshold value for the "Cosine Similarity" and the "Pearson's Correlation". Therefore only a few restaurants with high similarities are recommended.
The metrics used to evaluate the model is discussed in the next section.

IV. Results

A. Comparing Pearson's Correlation, Cosine Similarity and Euclidean Distance

```
recommend_2('Meghana Foods','Cosine')
```

name	rate	location	dish_liked	cuisines	approx_cost	neighbourhood	addr
Anand Donne Biriyani	3.6	Jayanagar	NaN	Biryani	200.0	Banashankari	Banashankari Jayanagar
Biryanis And More	4.0	Jayanagar	Prawn Biryani, Dragon Chicken, Chicken Boneles...	Biryani North Indian Chinese Andhra South Indian	750.0	Banashankari	Banashankari Jayanagar

Fig. 5. Output using Cosine Similarity

```
recommend_2('Meghana Foods','Pearson')
```

name	rate	location	dish_liked	cuisines	approx_cost	neighbourhood	addr
Anand Donne Biriyani	3.6	Jayanagar	NaN	Biryani	200.0	Banashankari	Banashankari Jayanagar
Biryanis And More	4.0	Jayanagar	Prawn Biryani, Dragon Chicken, Chicken Boneles...	Biryani North Indian Chinese Andhra South Indian	750.0	Banashankari	Banashankari Jayanagar

Fig. 6. Output using Pearson's Correlation Coefficient

```
recommend_2('Meghana Foods','Euclidian')
```

name	rate	location	dish_liked	cuisines	approx_cost	neighbourhood	addr
Anand Donne Biriyani	3.6	Jayanagar	NaN	Biryani	200.0	Banashankari	Banashankari Jayanagar
Biryanis And More	4.0	Jayanagar	Prawn Biryani, Dragon Chicken, Chicken Boneles...	Biryani North Indian Chinese Andhra South Indian	750.0	Banashankari	Banashankari Jayanagar
Vindu Andhra Ruchulu	3.8	Jayanagar	Raita, Chicken Curry, Fish, Mutton Biryani, Be...	Biryani North Indian Andhra	800.0	Basavanagudi	Basavanagudi Jayanagar
Desi Rasoi	3.8	Jayanagar	Hara Bhara Kebab, Lunch Buffet, Naan, Babycorn...	Biryani North Indian Chinese Rajasthani	600.0	Basavanagudi	Basavanagudi Jayanagar

Fig. 7. Output using Euclidian Distance

As we can observe from the images Figure 5, Figure 6, Figure 7 the recommendations from the models using different distance or similarity metrics output similar results for a given user's preference. This is observed as we are considering finite amount of data to vectorize and compare the similarity thus we observe a similar results across various approaches.

Before evaluating the model, the data set was shuffled to allow a fair partitioning of the data into train and test splits. For the evaluation, 5000 samples were used as training data and 1000 samples were used as testing data. During evaluation, the criteria used for filtering a recommendation are indicated in Table II.

TABLE II
CRITERIA COMPARISION

Model	Criteria	
	TF-IDF	Neural-Network
Cosine Similarity	>0.7	>1.0
Pearson's Correlation Coefficient	>0.7	>0.99995
Euclidean Distance	<1	N/A

The restaurant recommended by the model were evaluated using the following criteria:-

B. Coverage

The Coverage of a recommender system is defined as the ratio of the number of training instances predicted by the model to the total length of the training data when the test data is used as queries.

TABLE III
COVERAGE COMPARISON

Model	Coverage	
	TF-IDF	Neural-Network
Cosine Similarity	0.4212	0.2922
Pearson's Correlation Coefficient	0.4208	0.774
Euclidean Distance	0.6122	N/A

C. Quality

The Quality of a recommender system is defined in terms of the average number of predictions made and the similarity of these recommendations with the test instance.

TABLE IV
AVERAGE PREDICTIONS COMPARISON

Model	Average Number of Predictions	
	TF-IDF	Neural-Network
Cosine Similarity	8.52	8.278
Pearson's Correlation Coefficient	8.511	30.396
Euclidean Distance	15.848	N/A

D. Inferences

A few inferences observed during the exploratory data analysis phase and the training/testing phase are as follows:-

- Since the location of a restaurant was considered as a parameter for recommendation, it can be stated that restaurants which are located closer to other popular restaurants and areas are more likely to be visited often by the user. Therefore, new restaurants prefer to have their branches in well known areas in spite of increasing expense.
- Restaurants offering a variety of cuisines tend to be more popular. However, offering too many cuisines can mean that the restaurant may not be prioritizing the quality of food/service. This can significantly impact their ratings. Hence, restaurants need to maintain a balance between number of cuisines offered and quality of food.

E. Drawbacks

The approach taken and models built have the following drawbacks/failure cases:-

- The attributes taken into consideration to compute similarity measures between restaurants are 'location', 'ratings', 'cuisines' and 'dishes-liked'. This is done because the other attributes are either irrelevant or contain many NaN values and cannot be used for recommendation.
- The number of restaurants recommended varies for each query instance as the thresholds for similarity and distance measures are set beforehand. This means that each query instance can have too many or too few restaurants recommended. This directly effects the coverage and quality of the recommender system.

The future scope of the project can involve

- Using a collaborative filtering method to predict the exact rating for a user. This will help to improve the quality of recommendations.
- Storing of user reviews and ratings will help to build up a user profile. This can be then used to make the recommendations more personalized.

V. CONCLUSION

This project aimed at delivering a Recommender System for restaurants using multiple approaches as well as analyzing the wide variety of recommender systems that have already been implemented. A bunch of research papers majored in the field of restaurant recommendations were discussed in this paper. 3 different approaches for recommending restaurants are presented. These approaches are thoroughly explained with their respective benefits and shortcomings. Using Location, Cuisine, and Dishes-Liked data, accurate recommendations are made. But, there is still scope for improvement by incorporating sentiment analysis or text analysis on reviews to explore hidden aspects of user reviews. Most of the recent recommender systems try to extract criteria from unstructured user reviews to provide more personalized recommendations to specific user.

In conclusion, this project provides a comprehensive analysis of the various attributes that helps a user choose certain restaurants over others. various attributes that helps a user choose certain restaurants over others.

A. Member Contributions

- **Rahul D Makhija**:Pearson Correlation, Euclidean Distance, Cosine Similarity Model and Model Evaluation.
- **Rohan M**: TF-IDF - Cosine Similarity Model and Visualization
- **Rishab K S**: Neural Network Model and Prepossessing.

All of us referred to several research papers and contributed towards Literature Survey

REFERENCES

[1] Sumaia Mohammed Al-Ghuribi and Sharul Azman Mohd Noah, " Multi-Criteria Review-Based Recommender System - The State of the Art "

[2] Gao, Yifan, et al. "A restaurant recommendation system by analyzing ratings and aspects in reviews." International Conference on Database Systems for Advanced Applications. Springer, Cham, 2015.

[3] J. Priya, "Predicting Restaurant Rating using Machine Learning and comparison of Regression Models," 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), 2020, pp. 1-5, doi: 10.1109/ic-ETITE47903.2020.238.

[4] Dwivedi, Prerna, and Nikita Chheda. "A hybrid restaurant recommender." International Journal of Computer Applications 55.16 (2012).

[5] A. Sarkar, A. Baksy and V. Kirpalani, "Analysis of Zomato Services using Recommender System Models," 2021 International Conference on Intelligent Technologies (CONIT), 2021, pp. 1-5, doi: 10.1109/CONIT51480.2021.9498534.