Restaurant Recommendation System in Dhaka City Using Machine Learning Approach

Abstract- Due to the huge explanation of artificial intelligence, machine learning technology is being used in various areas of our day-to-day life. In Dhaka city, there are lots of Restaurants. Sometimes many of us face a scenario where we go to a restaurant and order some food but the food is not that good or the price of the food is very high as compared to other restaurants. Apart from this, another major problem is the location of the restaurant. We cannot find the best restaurants around us based on our preferences. This research proposed a model by using a machine learning algorithm that will be able to suggest a suitable restaurant based on the user's criteria. We have collected Dhaka city's restaurant's data from various websites, then we have used Weight-based score calculation and cosine similarity matrix to build our machine learning model. This recommendation system will also suggest similar restaurants based on the user's selected restaurants.

Keywords— Recommendation System, Content-Based filtering, Machine Learning, Dhaka City.

I. INTRODUCTION

A recommendation system is a model which can predict user interest and suggests the desired item according to the user's preference [1]. Due to the rapid increase of internet users, the recommendation is being used on many platforms nowadays. Many e-commerce websites and other well-known platforms like Netflix, LinkedIn, Facebook, Instagram, and YouTube are using recommendation systems to give better results to their users [2]. Because of rapid development in Bangladesh's economy, as the capital of the country, Dhaka city is developing rapidly. Due to this progress, lots of restaurants have opened their business in Dhaka city. Now there is a situation where there are 8 -10 restaurants in one building. However, the problem is among these restaurants very few are providing good services so it's become a dilemma for food lovers to choose quality restaurants [3]. This research is to try to encounter this problem with a machine learning model to recommend restaurants for the users based on their criteria.

Our main intention is to give users their desired restaurant according to their preferences like Location, Cuisine type, price range etc. This machine learning model works in two different phases. In the first stage, the model will take user input for user preferences then calculate it as a score for every restaurant that is stored in our database. After getting the score it will sort the restaurants based on the score and give the best 10 restaurants that match with the preference. When users choose any restaurant we started to work on our second stage of the model in the background. We take the restaurants that users have chosen and use another machine learning algorithm cosine similarity to figure out the most similar restaurant which user chooses by using content-based filtering [4].

This recommendation system will play an important role in terms of user satisfaction because according to the experts Dhaka city has over 10000 restaurants in different locations and every restaurant has its different environment, food culture so it will help users to find out their suitable restaurant without going to the restaurant. Due to the huge expansion of the internet, there are more than 45 million internet users in Bangladesh. Hence, it will be helpful for the users to give their preferences from home on the internet and the model will suggest the best restaurant nearby the user.

In this research, a new model is proposed where the model deals with two different approaches and merges the best result. Firstly, it takes user input like location, price range, rating, cuisine type. Secondly, it calculates the score based on weight where 90 percent consist of user ratings and 10 percent consists of the price range. After getting the score the model will sort the restaurants based on the score in non-decreasing order and suggest the user top 10 restaurants. Lastly, by taking users chosen restaurants the model uses another machine learning approach content-based filtering, and suggests similar restaurants that users have chosen. So it will give a great overview to the users of which restaurants are best among these huge numbers of restaurants in Dhaka city.

II. LITERATURE REVIEW

Many works have been done in the recommendation system [5-8]. However, none of them are specifically based on Dhaka city's restaurant. Our paper is mainly based on the city of Dhaka, to do so we have to review some previous work on the recommendation system. Some of them are given below and described shortly.

A paper published in March 2020 has been published by the author that works in three phases: learning, analyzing, and recommendation. There were about 10000 data collected from the yelp website. They have used collaborative filtering to find similar users based on user preference and then recommend those restaurants that similar users like. So whenever a user searches on that system it shows the right rating restaurants [5].

Another paper published in 2016. In this paper, the authors proposed a location-based restaurant recommendation system for users that asked for data from the user like his

preferences (area, restaurant type, etc.) and user location. Then by using Cosine Similarity they figure out restaurants that match with user preference and calculate the distance between the user and all the restaurants. After that, make a weight-based score that recommends the best restaurant [6].

$$Score = A * Similarity + B * Distance \ldots \text{(i)}$$

In 2015 a paper was published. The authors of this paper addressed some interesting techniques to build a hybrid recommendation system. The system will take some psychographic information from the user such as the number of calls per day, mobile phone use per day, internet user per day, and calculate a psychological score to determine how social the user is. They call it a collaborative score. After this, the system also takes some demographic information as well like, income, age, gender, and calculates a demographic score. After this, they figure out the net score considering these two parameters and recommend restaurants based on this score [7].

$$Net\ score = (x * collaborative\ score) + ((1 - x) * demographic\ score) \ldots\ldots \text{(ii)}$$

Some authors published a conference paper in 2013 with the titled Location-based personalized restaurant recommendation system for mobile environments. In this paper, they have proposed a restaurant recommendation system for web applications. They used the foursquare web to retrieve users past visits to restaurants. If the user does not visit any restaurant, then the system tries to retrieve the user's friend's last visit to the restaurant. In the backend, the model calculates a score of every restaurant by using a naive biased algorithm and recommends it to the users in the frontend. So when the user moves from one place to another the model constantly shows different restaurants based on score. [8].

Apart from that restaurant recommendation system, there are some other famous recommendation systems that we have gone through to get a better understanding of how to build a recommendation system. A paper was published where the author has made a recommendation system by using the k-means clustering algorithm and k nearest neighbor to recommend a movie to the user. First of all, the system takes input from users like name, gender, age, occupation and tries to put this information on a cluster through utility metrics. After applying the k nearest neighbor algorithm they recommend a movie that matching cluster's people watch, like, and given a good rating [9].

III. PROPOSED METHODOLOGY

In this section, we will briefly discuss the whole process that we did to build this restaurant recommendation system. From data collection to building the actual model, we had to come through lots of stages. In this section, we will elaborate on data collection, data processing, feature selection, algorithm building phases.

A. Data collection

There are over 10000 restaurants in the city of Dhaka. Hence, it was quite impossible to go to every restaurant and collect data from the users. We took an alternative approach. We use various websites like Google Maps, Facebook, and Food panda to collect data. At the initial stage, there were about 3000 restaurants data. But due to collecting from various sites, there were lots of issues with the dataset. We used various data processing techniques to shape our data and make it usable for our model.

B. Data processing

After getting the raw data from the data collection step we had to process the whole dataset and make it perfect for our model to train. First of all, we have to eliminate irrelevant data. Secondly, we deleted the missing and duplicate data, to do so we had to use various python tools like pandas, Matplotlib etc. After cleaning the dataset, we got almost 1600 restaurants data to work with our model.

C. Feature selection:

Feature selection is one of the most important parts of making a machine learning model because if we choose you features wisely our model will suggest restaurants accurately [10]. Our target level is the restaurant's name but we need to select the input level carefully. Our input features are location, cuisine type, price range, ratings, and number of ratings. We also consider the number of ratings as a level because a restaurant might have good ratings but the problem is very few people give those ratings so we cannot say that that restaurant is good or bad so that's the reason we consider several ratings as an input level.

Feature Name	Types	Domain
Restaurant Name	String	<Name>
Restaurant Address	String	Banani, Dhanmondi etc.
Ratings	Float	(0-5.0)
Restaurant Type	String	Fast food, Chinese etc.
Number of Ratings	Integer	(>=0)
Price Range	Integer	(1,2,3)

Table 1: Features that used to build restaurant recommendation system.

D. Data scaling:

We know that our computer does not understand anything except numbers. Hence, we have to convert a string value into a number. For this, we used a level encoder from the python Sklearn package. After converting the data, we used another Sklearn package MinMax scaler to scalar our data so our data's value was in the range of 0 to 1. We used this tool because when we used mathematical calculation it might cause miscalculations if we do not scale every feature on the same scale. So, to play safe we scaled it with a min-max scaler.

E. Model building:

After preparing the dataset now the most important come to build the actual recommendation model for our user. In recent time most of the recommendation system or model is being building by using Hybrid recommendation system [11]. We divided our model into two parts, one is a weight-based score and the other is content-based selection.

a) Weighted based score calculation:

A weighted-based recommendation system is the simplest yet a good way to recommend the best restaurant, it easily calculates a score for every restaurant based on their weighted [12]. The main aim of this method is to calculate an indexing score for every restaurant and sort it in non-decreasing order. Whichever restaurant gets the highest score that means that the restaurant has a good reputation. To calculate the average score we need to consider some features so we choose every restaurant's ratings and several ratings to calculate the average rating scores.

$$average\ rating = \frac{(R*NR)+(RM*NRQ)}{(NR+NRQ)} \ldots (iii)$$

Where,

R = ratings of a restaurant.
NR = number of people who give ratings
RM = means of rating column.
NRQ = 60% quantile of several ratings.

$$average\ price = \frac{3}{P} \ldots (iv)$$

Where,

P = price of that restaurant where 1 = cheap,
2 = medium, 3 = expensive.

After calculating the average ratings and average weight for every restaurant we are good to go to calculate the score for every restaurant. We have considered 90% of the average ratings and 10% of the price to calculate our score.
The equation is:

$$score = (average\ rating * 0.90 + average\ price * 0.10) \ldots . (v)$$

After getting the score for every restaurant we can now suggest the best restaurant according to the score. All we need to do is take the user preferences as user input and search them into our database. All the restaurants that will come we index them according to their score and we will get the best restaurant at the top.

b) Content-based similarity calculation:

After showing the users top restaurants according to their preference, if the user selects any restaurant the model will start to implement its second recommendation system which is content-based filtering. Content-Based filtering is one of the most popular recommendation systems that many well-known companies use to recommend their product to customers. Content-based filtering finds out similarities among the restaurants and which restaurant matches more with the user's chosen restaurant that will recommend first [13]. An example of content-based recommendation is given below for better understanding,

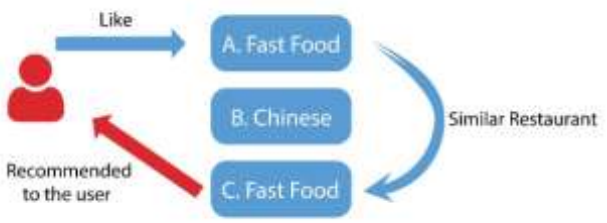

Figure 1: demonstration of content based filtering

Here we can see a user has liked a restaurant which main cuisine is Fast food and there are two other restaurants one is Chinese and another one is Fast Food. Hence, restaurant C (Fast food) is more similar to restaurant one as compared to restaurant B (Chinese). That's the reason restaurant one will be recommended to the user in content-based filtering.

Algorithm building:

There are lots of ways to calculate similarities of two products among them cosine similarity and Euclidean distance is vastly used [14]. In our system, we have used cosine similarity to calculate the similarity of all restaurants with every restaurant. We have used area, cuisine type, ratings, price range, and number of ratings as our variables to compare with.

$$\cos\theta = \frac{p.q}{\|p\|\|q\|} = \frac{\sum_{i=1}^{n} p_i . q_i}{\sqrt{\sum_{i=1}^{n} {p_i}^2} * \sqrt{\sum_{i=1}^{n} {q_i}^2}} \ldots (vi)$$

Here,

p and q are the features of the restaurant. Such as location, restaurant type, price etc.

The cosine similarity function will generate a similarity metric that has the similarity value of every restaurant with each other [15]. The value of all cells in this metric lies between zeros to one. We know that cos 0 degree is equal to 1 so if the value of restaurant[A][B] is close to 1 that means restaurant A is similar to restaurant B so we can suggest restaurant B who chooses restaurant A. So if a restaurant matrix value is close to 1 that means the restaurants are similar and if the value is close to 0 that means restaurants are not similar.

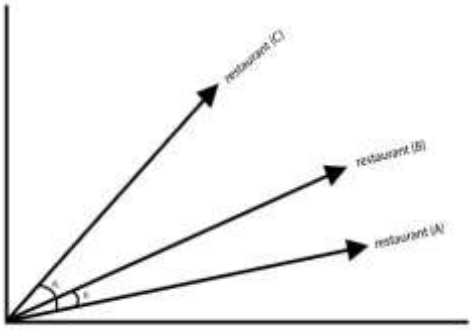

Figure 2: Similarity between three restaurants.

So when a restaurant has been selected by the user the system will go to the similarity matrix and extract the top 10 restaurants of that particular restaurant. Hence, those 10 restaurants will be recommended to the user. And that's the all working methodology of our restaurant's recommendation system.

IV. RESULT AND DISCUSSION

Now, our system is ready to recommend restaurants. We can implement this model in any web application or mobile application to achieve better results. There are lots of food delivery apps in Bangladesh like Patho, food panda, and so on those run businesses in Dhaka city. This model will increase their suggestion and performance to sell more items. First of all, the user will give this preference such as location, price range, food type, ratings then our model will filter the dataset based on score and give the user the best recommendation when the user choose one restaurant among them the model will work again and apply content-based filtering to find out similar restaurants and recommend it to the users.

To build content-based filtering we have used Cosine similarities, Euclidean distance, and Laplacian distance to calculate the similarity of two restaurants. We have found cosine similarity is more accurate in this case as compared to others algorithms.

An overview of our recommendation system is given below:

Step1: Take input of user's preference

```
enter restaurant area: Banani
enter restaurant type: Fast Food
cheap = 1
medium = 2
expensive = 3
enter restaurant price: 2
```

Step2: find out top restaurants based on user preference

Out[19]:

	index	Restaurant Name	Area	Restaurant type	price	Ratings	score	Number of ratings
0	156	Takeout Banani	Banani	Fast Food	2	4.4	4.695863	5232
1	88	Madchef Banani, Dhaka	Banani	Fast Food	2	4.3	4.001048	2759
2	87	Herfy Banani	Banani	Fast Food	2	4.2	3.912272	1786
3	143	BFC	Banani	Fast Food	2	4.1	3.833887	1941
4	90	Urban Spoon	Banani	Fast Food	2	4.1	3.828665	546
5	47	KFC Banani	Banani	Fast Food	2	4.0	3.753862	1252

Step3: After chosen any restaurant from input one recommend similar types of restaurant:

Out[21]:

	Restaurant Name	Area	Restaurant type	price	Ratings	Number of ratings
87	Herfy Banani	Banani	Fast Food	2	4.2	1786
143	BFC	Banani	Fast Food	2	4.1	1941
121	KFD Express	Banani	Chinese	2	4.3	1475
47	KFC Banani	Banani	Fast Food	2	4.0	1252
21	Fools Diner	Banani	Chinese	2	4.4	1399
146	Sajna Restaurant	Banani	Indian	2	4.1	1709
249	KFC Banasree	Banasree	Fast Food	2	3.9	1650
124	Soi 71	Banani	Chinese	2	4.2	895
90	Urban Spoon	Banani	Fast Food	2	4.1	546
156	Takeout Banani	Banani	Fast Food	2	4.4	5232

V. CONCLUSION

Due to the huge development of Dhaka city, many restaurants have been opened all over the city and the number of mobile phone users is increasing day by day. In this case study, we try to find out the best solution for users to find out his/her desirable restaurants with the help of this mobile phone. Despite having some limitations such as we don't have a similar type of user's data to build a recommendation system that suggests restaurants based on user behavior we tried to build a system where the system suggests restaurants based on user's review and restaurant similarities. If we implement it in our mobile sets through the web or apps, it would give us desired results before going to an unknown restaurant. It is very convenient for the users because there are lots of middle-class families in Dhaka city who can afford restaurants' money once or twice a month. Hence, it is very important to have prior knowledge about the restaurant before entering.

In the near future, we have plans to extend this project furthermore. We will implement this model into a web app and we will collect user's review data who used this system and classify them based on their age. As a result, we can suggest restaurants based on user behavior and user similarities based on the data set.

REFERENCES

[1] Nagarnaik, Paritosh, and A. Thomas. "Survey on recommendation system methods." In *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, pp. 1603-1608. IEEE, 2015.

[2] Walunj, Sachin Gulabrao, and Kishor Sadafale. "An online recommendation system for e-commerce based on apache mahout framework." In *Proceedings of the 2013 annual conference on Computers and people research*, pp. 153-158. 2013.

[3] Al-Rifat, Ahsan, and Rezwana Tasnim. "Factors influencing customers' selection of restaurants in Dhaka, Bangladesh." *The Business & Management Review* 10, no. 5 (2019): 186-198.

[4] Gigimol, S., and Sincy John. "A Survey on Different Types of Recommendation Systems." *Engineering and Science* 1, no. 4 (2016): 111-113.

[5] Varatharajan, Mr N., J. Guruprasad, and K. Mathumitha. "Restaurant Recommendation System Using Machine Learning." *International Educational Applied Research Journal* 4, no. 3 (2020): 1-4.

[6] Zeng, Jun, Feng Li, Haiyang Liu, Junhao Wen, and Sachio Hirokawa. "A restaurant recommender system based on user preference and location in mobile environment." In *2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, pp. 55-60. IEEE, 2016.

[7] Katarya, Rahul, and Om Prakash Verma. "Restaurant recommender system based on psychographic and demographic factors in mobile environment." In *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, pp. 907-912. IEEE, 2015.

[8] Gupta, Anant, and Kuldeep Singh. "Location based personalized restaurant recommendation system for mobile environments." In *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 507-511. IEEE, 2013.

[9] Ahuja, Rishabh, Arun Solanki, and Anand Nayyar. "Movie recommender system using K-Means clustering and K-Nearest Neighbor." In *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 263-268. IEEE, 2019.

[10] Gandhe, Ashish. "Restaurant recommendation system." *cs229. stanford. edu* (2015).

[11] Das, Debashis, Laxman Sahoo, and Sujoy Datta. "A survey on recommendation system." *International Journal of Computer Applications* 160, no. 7 (2017).

[12] Himel, Md Tayeb, Mohammed Nazim Uddin, Mohammad Arif Hossain, and Yeong Min Jang. "Weight based movie recommendation system using K-means algorithm." In *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 1302-1306. IEEE, 2017.

[13] Shah, Lipi, Hetal Gaudani, and Prem Balani. "Survey on recommendation system." *International Journal of Computer Applications* 137, no. 7 (2016): 43-49.

[14] Singh, Ramni Harbir, Sargam Maurya, Tanisha Tripathi, Tushar Narula, and Gaurav Srivastav. "Movie recommendation system using cosine similarity and KNN." *International Journal of Engineering and Advanced Technology* 9, no. 5 (2020): 556-559.

[15] Mansur, Farhin, Vibha Patel, and Mihir Patel. "A review on recommender systems." In *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, pp. 1-6. IEEE, 2017.