An Optimized Reactive Personalized Restaurant Recommendationss System Using Content-Based and DBSCAN Clustering in Google Cloud Platform

Nadia F. Al-Bakri[1(✉)], Farah Saad Al-Mukhtar[1], and Faten Alabbas[2]

[1] Department of Computer Science, College of Science, Al-Nahrain University, Jadriya, Baghdad, Iraq
{nadia.f.al-bakri,farah.saad}@nahrainuniv.edu.iq

[2] School of Graduate, Online and Continuing Education, Fitchburg State University, Fitchburg, MA, USA
Falabbas@fitchburgstate.edu

Abstract. Recommendation systems play a vital role in food industries. They provide a personal recommendation to customers based on their preferences and interests. Due to the diversity of restaurant types and services, users struggle to find relevant options. This challenge is further exacerbated by cold-start and long-tail problems, which cause the limitation of existing recommender systems. Personalized service recommendations will help users to identify and select appropriate restaurants according to their location and preferences. To achieve this objective, this research suggests a context-aware restaurant recommender system for personalized restaurant suggestions leveraging customer's profile preferences collected explicitly from mobile or web apps utilizing Firebase. It employes reactive content-based filtering combined with DBSCAN (Density-Based Spatial Clustering of Applications with Noise) clustering to enhance dining services through leveraging GPS API Google cloud platform for scalability and minimum computations. Furthermore, the core innovation lies in final restaurant ranking, which applies a proposed optimized weighted score formula utilizing cosine similarity measure and Manhattan distance with regularization coefficient on user and restaurant features, including location. Data was collected from customers around AL NAHRAIN university in Baghdad. Experimental findings show beneficial results for achieving precision, recall and F1-score of 0.95,0.92 and 0.93 respectively, showing effective support in customer decision-making.

Keywords: Content-based · Cosine Similarity · DBSCAN · Google Cloud,GPS-API

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2027
O. Gervasi et al. (Eds.): ICCSA 2026 Workshops, LNCS 16758, pp. 135–154, 2027.
https://doi.org/10.1007/978-3-032-30497-1_9

1 Introduction

A recommendation system is a tool that helps users to find relevant items by filtering and suggesting products or information they may be interested in. Its main goal is to reduce information overload and present users with a personalized list of suggestions. Recommendation systems are broadly classified into two main types: Collaborative Filtering (CF), which leverages patterns among multiple users' behaviors, and Content-Based Filtering (CBF), which focuses on analyzing attributes of items that users have previously liked. These approaches form the foundation for delivering tailored recommendations and are continuously refined to improve accuracy and relevance for consumers [1, 2]. Recommendation system for restaurants is an intelligent technique that provides users with vast lists of available dining options, saving their time and effort while providing assurance of satisfaction in food experience. various factors are considered such as ambiance, type of cuisine, food price, location, dietary restrictions, and user preferences, these systems suggest tailored recommendations that align with individual tastes [3]. In recent years, there has been a growing interest in location-aware restaurant recommendation systems. This area of research has gained attention due to its potential to improve user decision-making by providing real-time suggestions tailored to personal preferences and physical location. With the increasing reliance on mobile applications and their rapid development for decision-making, especially in the hospitality and food service industries, restaurant recommendation systems (RRSs) have become vital tools for enhancing customer satisfaction in finding relevant items [4]. For mobile and web development, Firebase is a platform suggested by Google that provides a suite of services, including real-time databases (Cloud Firestore and Realtime Database) [5]. Firebase enables rapid development and easy scaling of applications, offering front end and backend services like real-time syncing, authentication, and cloud storage with minimal server-side code [6].

Collaborative filtering [7, 8], are dominated by user-item interaction history which fail to work effectively in cold start or data sparse conditions, Conversely, content-based approaches exploit item properties and user preferences and, in this way, personalized can be given without a lot of user history [9] .Context aware recommender systems are becoming more important, in the context of mobile and web-based services (e.g., food and hospitality domains). In urban environments, the geographical closeness of restaurants is essential to estimate the rational value of recommendation. Researches have shown that users consider close by points of interests more important than far away even though they may potentially offer unimaginable experiences specially in densely populated regions [10, 11] Thus, geospatial intelligence coupled with proximity measures gives more contextually relevant results. Considering system recommendations, two personalization services are used, they are: [12, 13]

1. A proactive recommendation system that forecasts and serves personalized suggestions to customers without explicit demands depending on user history.
2. Reactive systems afford recommendations only when triggered or prompted by users such as (finding " nearby restaurants "). In these services, the system responds to the user's conveyed needs at the instant of service supply. In this paper, this type of service is utilized.

Many previous studies have revealed the efficacy of restaurant recommendation systems that include factors such as food quality, pricing, and customer reviews. For example, research by [14] emphasizes the efficiency of collaborative filtering (CF) methods such as memory-based CF using item-based and user-based and model-based CF using bayes theorem [15]. Matrix factorization, sentiment analysis, tailor-made for the restaurant field. However, they often miss user preferences and multiple attributes beyond mere ratings. Also, by [16], system applied content-based filtering on static restaurant attributes (e.g. cuisine type, category, address metadata) and a similarity between restaurants using the cosine similarity metric is computed, then a recommendation to users based on their extracted preferences is suggested. However, these systems suggested restaurants according to user's preferences, they did not fully integrate dynamic user filtering attributes and GPS-based navigation services and locations into their models. Also, they faced cold start item problem because of insufficient restaurant information services. As a result, these issues reduce the accuracy of real-time geospatial proximity computation.

In response to these demands, this paper proposes a reactive location-aware restaurant recommendation system for restaurant discovery using content-based method based on cosine similarity measure and DBSCAN algorithm for grouping regions according to longitude and latitude of restaurants from GPS coordinates and google cloud platform utilizing Firebase. Also, for ranking, an optimized weighted score formula is suggested leveraging similarity results between customers and restaurants and their locations using the latitude and longitude. The proposed system is deployed on web based and mobile platform. It is assumed that the customers are in AL Nahrain university in Baghdad capital of Iraq and need a suggestion for nearby restaurants according to their preferences. The main contributions of this work are as follows:

- Proposing a scalable, reactive restaurant recommendation model utilizing content-based filtering method enhanced with customer-defined attributes for food and service type.
- Applying DBSCAN clustering method to enhance dining services through leveraging GPS services and Google cloud platform for scalability and minimum time.
- Proposing a ranking proximity formula utilizing cosine similarity and Manhattan distance values for recommendation with a regularization coefficient.

The rest of the paper is organized as follows: in Sect. 2 we present an overview and a comparison of related works on restaurant recommendation systems. In Sect. 3 we present the architecture, detailing the components of the proposed system and the algorithms used. In Sect. 4 we present the used datasets, experiments, results and discussions. Finally, Sect. 5 concludes the research and outlines potential future directions.

2 Related Work

This section reveals several works that are related to the subject matter. They are as in the following:

Nur Idalisa et al. in [17], a machine-learning approach is used to develop a recommendation system of a restaurant to Malaysian customers. The research identified the

difficulties of choosing an appropriate restaurant using Trip Advisor platform. Another work by Elham Asani et al., [18] proposed a context-aware recommender system that food predilection is extracted from customer comments. Then food names are clustered and their opinions are analyzed. As a final decision, the nearby open restaurants are recommended based on their similarity to user preferences. One more study of Henry Dyke A. Balmeo and Albert A. Vinluan in [19], developed an application that used GPS for small and medium enterprises (SMEs), to locate and rank restaurants and highlighted the best performance using social media riposte. The system administrator manufactured measurable reports and compilation using data extracted from web-based social media. The system utilized both web and mobile platform and helped users find a restaurant with good service and products by displaying ratings from previous customers. Additional research of Sathya Seelan K and. K. PrabhuIn in [20], presented restaurants recommendation system that used machine learning algorithms to help customers find restaurants and popular food items. The system is built using a Kaggle dataset and a web application with Python's Flask framework. It provided better experience to users in searching for a short amount of time and nearby location restaurant. In one more research work of Qusai Y. Shambour et al. in [21], proposed a recommender system for restaurant based on user preferences. it utilized multi-criteria ratings, including implicit similarities between users and items, permeated similarity between user and user/item reputation. furthermore, the study of Michelle Hartanto and Ditdit Nugeraha Utama in [22], developed an intelligent decision support model that suggest suitable restaurants. Seven parameters were embraced (the interest of customer, fee/budget, distance between customer and restaurant, savor rating, rating of cleanliness, facilities rating, and halal or nonhalal status). The demonstration involved 75 restaurants in Jakarta and eight customers. Finally, the research of Md Sayeem Rahman et al. [23], Applied K-means clustering to analyze restaurant's geospatial distribution using latitude, longitude, rating, cuisines and price data, as well as identified hotspots and underserved areas. Also, the popularity of regional trends in cuisine and pricing are highlighted. Also, visualization results through interactive maps and statistical charts are depicted. Table 1 shows a summary of related works..

Table 1. Summary of literatures review

ref. no.	Dataset	Method	Gap
[17]	2,496 datasets gathered from the TripAdvisor platform.	Term Frequency-Inverse Document Frequency (TF-IDF), Cosine Similarity	The internal conflict of the secondary data sources that were used, that affected the reliability and validity of the results.
[18]	collected from the TripAdvisor website for 100 users during the first 9 months of 2018	Semantic Approach, Sentiment Analysis, Similarity Calculation	dependance on old NLP resources with restricted evaluation metrics.

(*continued*)

Table 1. (*continued*)

ref. no.	Dataset	Method	Gap
[19]	data extracted from social media platforms and Google Maps API.	Natural Language Processing Sentiment Analysis	The study focuses only on restricted physical locations and prices.
[20]	Kaggle dataset	Exploratory Data Analysis (EDA), Popularity-Based Recommendation, Collaborative Filtering (CF), singular value decomposition (SVD)	the absence of real-world testing and scalability issues.
[21]	The Restaurants-TripAdvisor MC dataset,The Hotels-TripAdvisor MC dataset, The Yahoo! Movies MC dataset	Hybrid User-Item based Multi-Criteria CF Algorithm	The research limited by dataset scope depends on ratings.
[22]	Zomato website	fuzzy logic, cosine similarity distance, optimization	small dataset with limited size limited parameters.
[23]	geospatial restaurant dataset collected and structured into CSV format.	Machine learning approach () combined with geospatial analysis, statistical calculations, and visualization techniques.	The dataset scope was limited, K-means clustering algorithm constraints in geospatial clustering.

3 Methodology

The proposed context aware restaurant recommender system employed on both web and mobile applications built on the Google Cloud Platform that incorporates content-based and DBSCAN clustering approaches with an effective ranking formula for recommendation. The proposed framework is initiated considering different dynamic attributes such as customer and restaurant locations using GPS coordinates and google map information, food categories, and service types. These attributes are extracted and preprocessed and then combined to be utilized in a weighted score formula to recommend relevant restaurants for each customer. The essential purpose of the suggested model is to propose an optimized weighted score formula considering the similarity values of selected attributes (food types and service types) and the distance values. At the same time, the personalization experience is enhanced by introducing DBSCAN clustering to improve the quality of the recommendation.

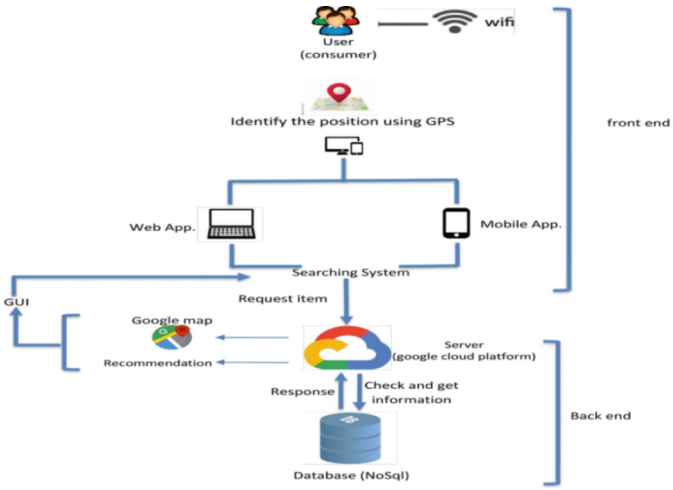

Fig. 1. Block Diagram of Proposed System Model

The suggested stages are especially helpful for customers, aiming to minimize decision-making time and maximize satisfaction. As mentioned earlier, the proposed system is designed to assist the customers in finding the nearest required restaurant according to longitude and latitude of each restaurant. It assumed that the target customers are at AL Nahrain university in Baghdad city in Iraq and they want to find the nearest restaurant using the proposed model. Additionally, the system provides a map display and distance calculations, and directional guidance to assist customers in navigating to their chosen restaurant.

The proposed model consists of two parts as shown in Fig. 1:

- The front-end is the process in which the contents are viewed over the application. The web/mobile restaurant recommender applications introduce services to customers through internet connection to request the favorite restaurant, after determining their current location from mobile GPS sensor. It allows the system to know exactly where the customer is, by determining the user's location which is determined by coordinates (longitude and latitude) using GPS technique that exists in either smart phone (iOS, Android) or PC by Wi-Fi (WPS) where their coordinates are automatically determined.
- The back end under Google cloud platform is the core of the proposed system that analyze and manage applications using Firebase as framework which helps in developing web and mobile applications for storing the database. NoSQL is the central piece for organizing and designing databases saving data as a pair of key-value.

3.1 Dataset Collection Stage

As mentioned earlier, restaurant locations were searched and collected using GPS, resulting a Dataset containing about (315) surveyed restaurants and can be expanded further. The attributes within the dataset are in the form of:

- Spatial Data: restaurant locations (coordinates latitude (X),longitude(Y)), they are provided by Google map service.
- Attribute Data: restaurant survey information that involves (restaurant name, address, rating, services type (Dine-in, Takeaway, Delivery, Drive-through) and food type (cuisine (eastern, western)). The service may differ from restaurants to another.

After dataset collection, Excel file records are filled and saved as a Comma-separated values (CSV file) then converted to JavaScript object notation (JSON) file by the online converter using the link https://www.convertcsv.com/csv-to-json.htm. The JSON format was chosen because it is quite compatible with Firebase which is commonly used for transmitting data in web applications. Then the created data is uploaded and stored in Google cloud platform by using firebase Console where each restaurant is stored as a JSON object with a restaurant ID. For an account creation in Google firebase, a sign-in step is needed via the site https://firebase.google.com. Then a payment account is created using visa card on Google cloud platform up to 12 months for free of charge to build and run the website, application and services, including firebase and the Google Map API for supporting Android's SDK. Then API key is generated to enable using map features in an interactive approach. In addition, the location of restaurants is located using Google map service and the Real-time database (NoSQL) is filled with the data.

3.2 Customer Profiling Stage

Creating a customer profile is an initial stage in the suggested context-aware restaurant recommendation system. A depiction of the customers' needs is built, by considering the preferences and geographic location. A profile for each customer is composed. It consists of three types of information:

1. Demographic information: It represents personal details like customer names. This data collected through registration form.
2. Preference information: It is formed employing user choices in restaurants including food information such type of services (Dine-in, Takeaway, Delivery, Drive-through) and food type (cuisine (eastern, western)). Users should specify preferences accurately.
3. Location information: It represents the customer's current location, mostly given as latitude and longitude coordinates using Google map.

3.3 Preprocessing Stage

In this stage, the attribute values within the dataset are preprocessed for further implementation by encoding restaurant categorical features into vectors of zeros and ones. Categorical features such as food types and service types are transformed to numerical representation to be prepared for use in content-based algorithm. This process involves the following:

a. Categorical features:

- Single-label categorical (e.g., eastern, western) using One-Hot Encoding.
- Multi-label categorical (e.g., services (Dine-in, Takeaway, Delivery, Drive-through) using Multi -Hot Encoding.

- Ordinal categorical (e.g, rating (from 1 to 5)) using Ordinal Encoding.

b. Represent each restaurant as a feature vector.
c. Represent customers in AL Nahrain university preferences as a query vector.

3.4 Implementation

This section demonstrates the implementation process of the proposed model. Initially, the customer's GPS API coordinates are acquired using Google Maps. For example, the current customer's location is at AL-Nahrain university in Baghdad as shown in Fig. 2. The process starts when the customer searches for restaurants, the proposed system queries the server and retrieves JSON data file from Firebase and displays a list of available restaurants.



Fig. 2. Customer location and Available Restaurants

Clustering with DBSCAN

As a next step, a clustering algorithm DBSCAN [24] is applied that group points (restaurants in our model) that are close together in space using restaurant coordinates(longitude, latitude). This step is being taken to reduces search space and ensure locality. The reason for choosing DBSCAN clustering method is that it requires no prior choice of cluster count, inherently handles biased shapes, and marks outliers as noise. The hyperparameter such as EPS (Neighborhood radius) is tuned to local density around Al-Nahrain University. Also, min_samples (Minimum points per cluster) are adjusted to avoid sparse clusters. After clustering process is done, number of clusters is obtained, and the nearest clusters are found whose centroids are closest to the customer's location coordinates.

Distance Computation

After that, the distance between the customer's location (in (AL Nahrain University) and the restaurant locations is found within the nearest clusters using Manhattan distance as depicted in Eq. (1). As was declared earlier, the suggested restaurants are restricted to a geographically bound urban area, therefore Manhattan distance is adopted as an efficient approximation for spatial nearness. This formula is abundant for relative ranking while minimizing computation complexity compared to great-circle distance calculations. Table (2) depicts a gained set of restaurants with their attributes and the calculated distances. In this table, it is assumed that the target customer is with the coordinates (Longitude X = 44.37753,Latitude Y = 33.27927).

$$\text{Distance}(u, r) = |lat_u - lat_r| + \left|long_u - long_r\right| \quad (1)$$

where target customer coordinate is (lat_u,$long_u$) and restaurant coordinate is (lat_r, $long_r$).

Table 2. Restaurant Information Data Base and Distance Results

restaurant name	Status	Services	rating	Longitude X	Latitude Y	Address	Distance
99 grill	11 AM- 2 AM	Dine-in,Takeaway, Drive-through	4.1	44.3711	33.37526	Al-Dhubbat Street – Al-Adhamiya	0.09236
Boba restaurant	12 PM-11:30 PM	Dine-in, Takeaway	4.2	44.3683	33.30225	Al-Kindi Street	0.01655
Chili House	10:30 AM- 12 Am	Dine-in, Takeaway	4	44.38951	33.28115	Al-Jadriya	0.01666
Dawa	8 AM- 1 AM	Dine-in, Takeaway	4	44.37789	33.28506	Al-Jadriya	0.00895
Fried Chicken	10 AM- 12 AM	Dine-in, Takeaway	4.2	44.40834	33.28942	Al-Jadriya	0.04376
Hardees	10 Am-2 AM	Dine-in, Drive-through	3	44.39012	33.28125	Al-Jadriya	0.01737
KFC	8 AM- 1 AM	Dine-in, Takeaway	3.9	44.39043	33.28129	Al-Jadriya	0.01772
Levant Baghdad	8 AM- 1 AM	Dine-in, Takeaway	4.1	44.39019	33.29455	Al-Karrada	0.03074
RoyalGalaxy	9 AM - 12 AM	Dine-in, Takeaway	3.9	44.36496	33.31493	Al-Kindi Street	0.02589
Asrar Adam	9 AM- 12 AM	Dine-in, Takeaway	4.1	44.38879	33.2895	Al-Jadriya	0.02429
Al-Haidar Khana	8 Am-12 AM	Dine-in, Drive-through	3.4	44.38751	33.28248	Al-Jadriya	0.01599
Saysban	7 AM–12 AM	Dine-in, Takeaway	3.8	44.39126	33.28312	Al-Jadriya	0.02038
Shmeisani	8 AM- 12 AM	Dine-in, Takeaway	4.1	44.40683	33.28859	Al-Jadriya	0.04142
Qasr Al-Karam	11 AM- 12 AM	Dine-in, Takeaway	4.3	44.37892	33.28585	Al-Jadriya	0.01077

(*continued*)

Table 2. (*continued*)

restaurant name	Status	Services	rating	Longitude X	Latitude Y	Address	Distance
Al-Jadriya Floating Restaurant	11 AM- 12 AM	Dine-in, Takeaway	3	44.37812	33.28652	Al-Jadriya	0.01064
Al-Rahman Traditional Food Restaurant	Only Breakfast	Dine-in, Takeaway	5	44.39354	33.29201	Al-Jadriya	0.03155
Al-Rayahin Family Restaurant	10 AM–12 AM	Dine-in, Takeaway	3.5	44.33706	33.38784	Al-Krayat Corniche	0.0709
Barley Restaurant	7 AM-3:30 AM	Dine-in, Takeaway	4	44.43858	33.30537	Street 42	0.08995
Set Al-Sham Restaurant	24 Hours	Dine-in, Takeaway	3.9	44.36716	33.37416	Al-Adhamiya	0.08732
Qasim Abu Al-Kass Restaurant	9 Am-2 AM	Dine-in, Takeaway	3.7	44.36359	33.36754	Al-Adhamiya	0.07713
Naseem Dijla Al-Khair Restaurant	12 PM- 1 AM	Dine-in, Takeaway	3.8	44.35712	33.39376	Al-Krayat Corniche	0.09688
Al-Jadriya Family Restaurant & Gardens	10 AM- 12 AM	Dine-in, Takeaway	4	44.39255	33.28471	Al-Jadriya	0.02326
		Customer Address		44.37638	33.27762		

Recommendation Generation

Similarity Computation.

Subsequently, after encoding restaurant categorical features: the type of services (Dine-in, Takeaway, Delivery, Drive-through) and food types cuisine (eastern, western) into vectors of zeros and ones to be prepared for use in content-based algorithm; a cosine similarity [24] is performed between customer preference vector and restaurant vectors within a designated cluster. The customer's preferences are prioritized according to the customer's choices. The cosine similarity equation between object u and object v is expressed as in the following formula:

$$similarity(u, v) = cosine\left(\overline{R_u}, \overline{R_v}\right) = \frac{\overline{R_u} \cdot \overline{R_v}}{\|R_u\| \|R_v\|} = \frac{\sum_{i=1}^{N} R_{u,i} \times R_{v,i}}{\sqrt{\sum_{i=1}^{N} \left(R_{u,i}\right)^2} \sqrt{\sum_{i=1}^{N} \left(R_{v,i}\right)^2}} \quad (2)$$

As shown in Table 3, cosine similarity is computed between the customer' vector and six restaurant attributes using Eq. (2).

Table 3. Similarity Results Between Restaurant Vectors and Customer Attributes

Restaurant Name	Rating	Dine-in	Drive-through	Takeaway	Delivery	Eastern	Western
Target user	5	0	0	1	0	0	1
Dawa	4	1	0	1	1	1	1
Al-Jadriya Floating Restaurant	3	1	0	1	1	1	1
Qasr Al-Karam	4.3	1	0	1	1	1	1
Al-Haidar Khana	3.4	1	1	0	1	1	1
Chili House	4	1	0	1	0	0	1
Hardees	3	1	1	0	0	0	1

Ranking and Recommendation

According to the contextual scenario of the proposed model, the ranking and recommendation process is based on generating relevant restaurants by cooperatively leveraging similarity values produced from applying cosine measure on the normalized attribute values and the distance similarity values between the target customer's preferences and restaurant's vectors. To ensure valid cosine computation, all attributes including rating values are normalized between [0, 1] using min-max normalization. for consistency with cosine similarity values, distance similarity is derived by normalizing distance values to insure they lie between 0 and 1. Accordingly, the following formula is utilized for distance similarity computation:

$$\text{Distance_similarity}_{\text{i}} = 1 - \frac{Distance_i}{\max_distance} \tag{3}$$

where distance_i is the resulted spatial distance from Eq. 1 and max_distance is the given maximum distance value. Next, the proposed optimized weighted score formula is applied to rank the required restaurants as shown in Eq. 4. A linear combination is implemented due to its often use in multi-criteria recommendations. Eq. 4 is as follows:

$$\text{Weight_score}_i = \alpha \cdot \text{Similarity}_i + (1 - \alpha) \cdot \text{Distance_similarity}_i - \lambda \cdot (\alpha - 0.5)^2 \quad (4)$$

where $\alpha \in [0, 1]$ is a tuning parameter to control the effectiveness of the spatial distance values and the content-based similarity. Following the experimental analysis, when a customer prioritizes near restaurants, the parameter *is set to* 0, in this case, restaurant's location will have an influential weight on the ranking score compared to restaurant attributes, revealing a cold-start item scenario. On the other hand, as α approaches to 1, restaurant attributes gradually contribute to the ranking score and subsequently influence the final decision-making process. In addition, λ is a regularization coefficient introduced as quadratic term $\lambda \cdot (\alpha - 0.5)^2$ that limits the optimization process by controlling extreme bias toward a single factor, thereby enhancing the robustness and generalization capability of the ranking model. After experiment iterations, the parameters α *and* λ are empirically set to 0.6 and 0.05 respectively providing a balance between the semantic relevance and spatial proximity. Table 4. depicts the normalized values for each attribute, along with the resulted values of calculating cosine similarity, distance similarity and the weighted scores. The final suggested restaurants are ranked according to the values of the weighted scores where the score values are sorted in descending order. Note: in Table 4. The superscript within each weighted score represents its rank.

Table 4. Cosine Similarity, Distance Similarity and the Weighted Score Values Result for Restaurants and their Ranks to Target Customer

	α	Rating	Dine-in	Drive-through	Take away	Delivery	Eastern	Western	Cosine Similarity	Distance similarity	Weight score
Target user	0.6	1	0	0	0.2	0	0	0.2			
Dawa		0.8	0.2	0	0.2	0.2	0.2	0.2	0.923913	0.999884	0.953801(3)
Al-Jadriya Floating Restaurant		0.6	0.2	0	0.2	0.2	0.2	0.2	0.874385	0.999862	0.924076(5)
Qasr Al-Karam		0.86	0.2	0	0.2	0.2	0.2	0.2	0.933134	0.999861	0.959325(2)
Al-Haidar Khana		0.68	0.2	0.2	0	0.2	0.2	0.2	0.851256	0.999793	0.910171
Chili House		0.8	0.2	0	0.2	0	0	0.2	0.971323	0.999785	0.982208(1)
Hardees		0.6	0.2	0.2	0	0	0	0.2	0.888888	0.99977	0.932743(4)

Deployment on Google Cloud

The online module works by getting the current user model data including the distance and the restaurant's preferences. Then designate the regions and assign weighted scores for each restaurant. After that, sorting the score values in descending order and generating top K = 5 relevant restaurants. Finally, Servers send the results to end user's device as shown in Fig. 3 displaying the result in list of restaurants including the details (photo, location, phone number) that show the 5 most restaurants as shown in Fig. 4 (a, b, c, d, e). Figure 5 shows the path between the target customer and the recommended restaurant.

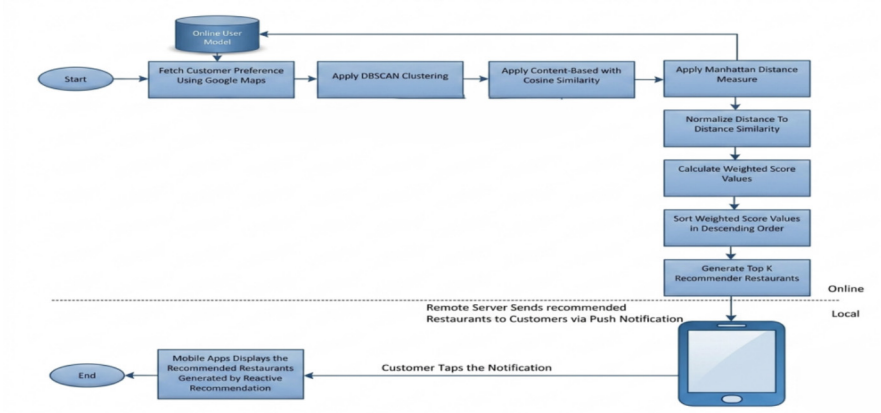

Fig. 3. The Flow Diagram of the Proposed Model

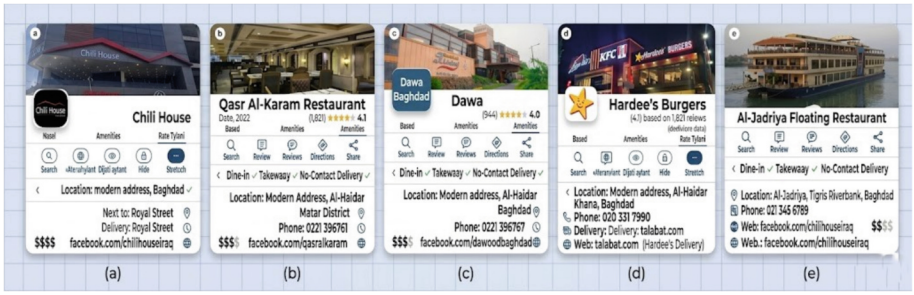

Fig. 4. The Recommended Restaurant (a) Chili House, (b) Qasr Al-Karam, (c) Dawa (d) Hardees (e) Al-Jadriya Floating Restaurant



Fig. 5. The Path to the Recommended Restaurant

4 Evaluation Metrics

In this section, the performance of the proposed context aware restaurant recommendations is evaluated using ranking recommendation-based measures. For each customer query, the proposed model generates a list of top K = 5 recommendations. For consistency and comparability, measures are calculated at a fixed top K = 5. The measures are defined as follows: [2]:

1. **Precision@K:** from the top K recommended restaurants, it measures the proportion of relevant restaurants among user's preferences.

$$\text{Precision@K} = \frac{Number\ of\ relevant\ restaurants\ in\ top\ K}{K} \tag{5}$$

2. **Recall@K:** from the recommended relevant restaurants, it measures the proportion of relevant restaurants that are successfully proposed in the top K.

$$\text{Recall@K} = \frac{Number\ of\ relevant\ restaurants\ in\ top\ K}{Total\ number\ of\ relevant\ restaurants} \tag{6}$$

3. **F1-score@K**: Balance mean of Precision@K and Recall@K.

$$\text{F1} - \text{score@K} = \frac{2 \times Precision@K \times Recall@K}{Precision@K + Recall@K} \tag{7}$$

4. **NDCG (Normalized Discounted Cumulative Gain)**: it is a rank-aware metric, that measures ranking quality by giving higher weight to relevant restaurants at the top of the list K.

$$DCG@K = \sum_{i=1}^{K} \frac{rel_i}{\log_2(i+1)} \tag{8}$$

$$NDCG@K = \frac{DCG@K}{IDCG@K} \tag{9}$$

where rel_i stands for relevance score of restaurants at rank i, and $IDCG\ @\ K$ stands for ideal DCG value.

5. **Coverage**: Percentage of restaurants that can be recommended by the system.

$$coverage = \frac{number\ of\ unique\ suggested\ restaurants}{total\ number\ of\ restaurants} \tag{10}$$

4.1 Discussion and Analysis

To ensure reproducibility and clarity, the experimental setup is explained as follows: the data was collected using the GPS API in Baghdad city. It comprises about 315 restaurants located in different regions around Al-Nahrain University. Restaurant profiles are created and stored in firebase cloud databases including restaurant name, status, services, longitude, latitude, rating and address. Also, considering customer's queries, a total of 30 customer profiles are retrieved from their explicit feedback via mobile or PC platforms for testing. At the same time, DBSCAN clustering method is configured within an average cluster radius of 0.7 km. The α in weighted score formula is set to 0.6 based on experimental tuning. To evaluate the proposed model, three baseline methods are utilized for comparisons, they are:

1. Distance based ranker: find recommended Top K restaurants using the Manhattan measure on distance attribute within 3 km.
2. Rating based ranker: find recommended Top K restaurants with highest rate.
3. Content based using TF-IDF (Term Frequency-Inverse Document Frequency) technique and cosine similarity with No-cluster Ranker: find recommended Top K restaurants applying TF-IDF with no clustering method.

Table 5. depicts a comparison of Precision@5, Recall@5, F1@5, NDCG@5, and Coverage across four conditions (Distance-only, Rating-only, No-cluster ranker, and the Proposed Model). The proposed model consistently outperforms baselines, particularly in ranking quality (NDCG) and balanced precision/recall as shown in Fig. 6.

Table 5. Comparison of Precision@5, Recall@5, F1@5, NDCG@5, and Coverage at K = 5

	Precision@5	Recall@5	F1@5	NDCG@5	Coverage
Distance based rank	0.72	0.66	0.69	0.75	0.48
Rating based rank	0.68	0.61	0.64	0.70	0.41
Content-based (no clustering)	0.80	0.77	0.78	0.76	0.56
Proposed model (DBSCAN with Content-based)	0.95	0.92	0.93	0.88	0.63

From the results shown in Table 5, it was concluded:

1. Using Distance-only criterion: recommendations have led to decrease in recall performance metric, since they recommend nearby restaurants that may not match user's requested preference services. In this case α is set to 0. Therefore, this case achieved location relevance at the expense of preference relevance.

2. Using Rating-only criterion: recommendations led to favor highly popular restaurants while ignoring contextual features such as distance and customer‘s preferences. This case may achieve better perceived quality based on ratings, at the expense of disregarding distance and services.
3. Using Content-based TF-IDF without employing clustering algorithm ranker criterion: recommendations have led to improve relevancy but with more time consuming, as it searched the entire regions without geographic filtering.
4. Using the proposed model (DBSCAN with cosine similarity measure): recommendations achieved the highest performance across all ranking metrics. principally in Precision@5 and NDCG@5, the results show that using cosine similarity on customer and restaurant features combined with DBSCAN clustering gained more related and geographically suitable recommendations. DBSCAN clustering effectively localized the search to high-density regions, while cosine similarity ensured better alignment with user preferences. Better Performance gained in Precision@5 and NDCG@5 indicated improved ranking quality and user potential satisfaction. Also, it was shown that the coverage metric within the proposed model was higher than distance only and rating-only baselines, demonstrating that the proposed context aware model avoids suggesting the same common restaurants. So, the proposed model balanced between locality, relevance, and quality, yielding moderate results and this reduced the risk of long tail problems, where only common restaurants are recommended.

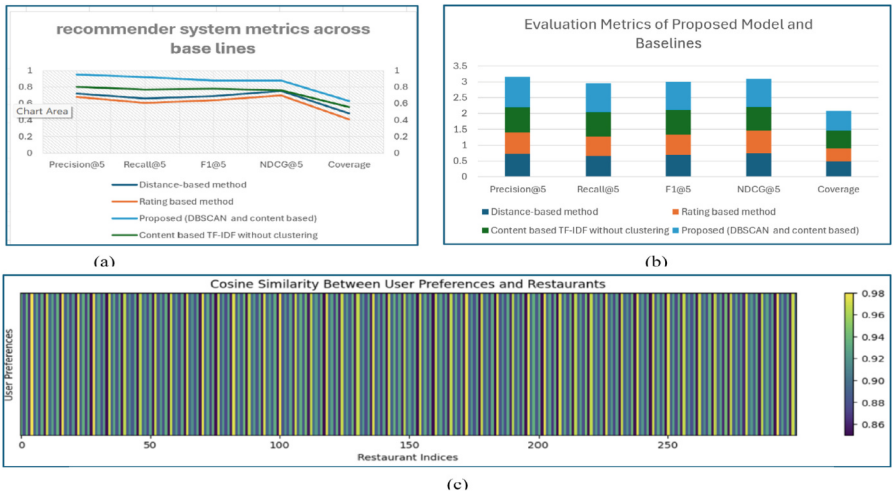

Fig. 6. (a) Line Plot of Metrics Across Recommender Baselines . (B) Recommended Baselines Across Evaluation Metrics. (C) Heatmap Visualization Showing the Relation Between Similarity Values. Darker Shades Indicate More Related Position with User-Requested Features.

Figure 7 illustrates a pie chart illustrating the distribution of cuisine types among the top 5 recommended restaurants around AL Nahrain University.

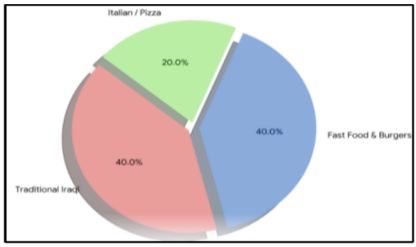

Fig. 7. Distribution of Cuisine Type

4.2 Impact of DBSCAN Clustering

In the section, the impact of restaurant-level clustering based on the restaurant location is discussed. It was depicted that the restaurants are spread into N clusters using DBSCAN clustering. This spread significantly improved both efficiency and recall. By restricting candidate restaurants to the nearest dense cluster, the system avoided irrelevant distant options while still capturing a diverse set of nearby restaurants. The inclusion of DBSCAN improved efficiency by reducing average candidate count from 315 to 61 per query i.e. 80.6% reduction in search space. The enhancement in Recall@5 reveals that using DBSCAN clustering methods benefits in capturing relevant restaurants within the customer's neighborhood. In addition, the coverage metrics also increased, showing that clustering prohibited the recommender system from frequently recommending the same few high-rated restaurants, i.e. long tail problem is avoided, which is a major problem in recommender systems. As a result, DBSCAN clustering was principally operative in inner-city environments like Baghdad, where restaurants are naturally grouped into dense regions. In Fig. 8, DBSCAN clustering scatter plot of restaurant locations, with clusters color-coded to highlight spatial grouping, each color represents a distinct cluster. Outliers are marked as noise points.

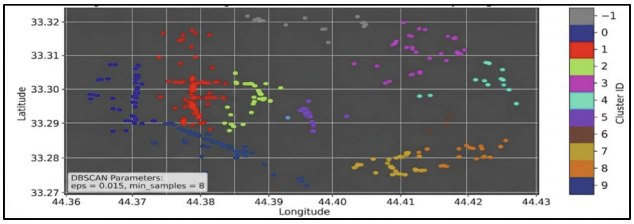

Fig. 8. DBSCAN Clustering Restaurants around Al-Nahrain University.

5 Conclusion

As a conclusion for this research, the findings confirmed that incorporating geospatial information for both the customers and restaurants significantly improves recommendation quality. By integrating GPS-API based distance factor with user preference

services, the proposed system efficiently suggests context-aware recommendations that reflect real-world user interests. Furthermore, combining DBSCAN clustering leveraging GPS services and google cloud platform noticeably reduced search space, leading to a remarkable decrease in computations. The core improvement in this work is built upon proposing a weighted score formula for ranking, which combines cosine similarity and distance similarity through two tunable regularization parameters. This formulation provides an effective trade-off between precision and recall, aligning with practical user expectations and adjusting spatial thresholds, offering a valuable insight for future system optimization and deployment in real-world restaurant recommendation scenarios. Experimental results yielded from experiments demonstrate that related restaurants are dependably ranked near the top of the recommendation list, as reflected by improved Precision@K, Recall@K, and NDCG@K values. Overall, a balance is achieved combining the similarity distance factor using GPS API based and content-based algorithm using cosine similarity to suggest relevant restaurants for customer preferences. The insights realized from adjusting spatial thresholds and weighting parameters provide valuable guidance for future recommender system optimization with large-scale deployment, and extension to other data set context-based recommendation fields.

Acknowledgments. This study was funded by nobody.

Disclosure of Interests. The authors have no competing interests.

References

1. Bahjat, M.I., AL-Bakri, N.F.: A Survey on semantic recommendation system based on linked open data (LOD). In: Book a Survey on Semantic Recommendation System Based on Linked Open Data (LOD), pp. 191–197. IEEE (2022)
2. Bahjat, M.I., AL-Bakri, N.F.: Accuracy and diversity enhancement of recommender system using linked open data. In: Accuracy and Diversity Enhancement of Recommender System Using Linked Open Data, p. 040003. AIP Publishing LLC (2024)
3. Nandan, M., Gupta, P.K.: Designing an efficient restaurant recommendation system based on customer review comments by augmenting hybrid filtering techniques. Universal Journal of Operations and Management, 15 (2023)
4. Karmakar, K.: A Context-Aware Approach to Restaurant Recommendations: System Algorithm and Case Study (2020)
5. Ahmed, E.Q., Ameen, Z.H., Al-Mukhtar, F.S., Jaaz, Z.A.: Maximizing Mobile communication efficiency with smart antenna systems using beam forming and DOA algorithms. In: Maximizing Mobile Communication Efficiency with Smart Antenna Systems Using Beam Forming and DOA Algorithms, pp. 453–460. IEEE (2023)
6. https://firebase.google.com/docs
7. Tuma, F.S., Ibrahem, N.F.: Hybridizing collaborative filtering with naive bayes classifier for cold-start recommender systems: enhancing movie recommendations in the age of big data. In: Hybridizing Collaborative Filtering with Naive Bayes Classifier for Cold-Start Recommender Systems: Enhancing Movie Recommendations in the Age of Big Data, p. 030009. AIP Publishing LLC (2025)
8. Ameen, Z.H., Al-Mukhtar, F.S., Ahmed, E.Q., Dhannoon, B.N.: A review of the past Decade's research outcomes on classifying cyberbullying. Al-Nahrain Journal of Science. **27**(1), 121–135 (2024)

9. Adomavicius, G., Bauman, K., Tuzhilin, A., Unger, M.: Context-aware recommender systems: from foundations to recent developments. In: Recommender systems handbook, pp. 211–250. Springer (2021)
10. Beirão, R.A.S.: A Multiple Criteria Route Recommendation System. ISCTE-Instituto Universitario de Lisboa (Portugal) (2020)
11. Sun, H., Wu, Z., Cui, Y., Deng, L., Zhao, Y., Zheng, K.: Personalized dynamic knowledge-aware recommendation with hybrid explanations. In: Personalized Dynamic Knowledge-Aware Recommendation with Hybrid Explanations, pp. 148–164. Springer (2021)
12. Ilarri, S., Trillo-Lado, R.: An approach for proactive mobile recommendations based on user-defined rules. Expert Syst. Appl. **242**, 122714 (2024)
13. Zhang, B., Sundar, S.S.: Proactive vs. reactive personalization: can customization of privacy enhance user experience? International journal of human-computer studies. **128**, 86–99 (2019)
14. Alabduljabbar, R.: Matrix factorization collaborative-based recommender system for Riyadh restaurants: leveraging machine learning to enhance consumer choice. Appl. Sci. **13**(17), 9574 (2023)
15. Ameen, Z.H., AL-Bakri, N.F., Al-zubidi, A.F., Hashim, S.H., Jaaz, Z.A.: A new COVID-19 patient detection strategy based on hidden Naïve Bayes classifier. Iraqi Journal ofScience, 6705–6724 (2024)
16. Christyawan, F., Rohman, A.N., Hartanto, A.D.: Application of content-based filtering method using cosine similarity in restaurant selection recommendation system. Journal of Information Systems and Informatics. **6**(3), 1559–1576 (2024)
17. Idalisa, N., Hazhar, M.H.M., Muslim, N., Albashah, N.L.S.: Restaurant recommendation system in Malaysia using machine learning approach. In: Electronics, Communications and Networks, pp. 319–325. IOS Press (2024)
18. Asani, E., Vahdat-Nejad, H., Sadri, J.: Restaurant recommender system based on sentiment analysis. Mach. Learn. Appl. **6**, 100114 (2021)
19. Balmeo, H.D.A., Vinluan, A.A.: Hanapresto: a platform for restaurant businesses with recommender system using knowledge extraction from social media and customer preferences. Asia Pacific Journal of Multidisciplinary Research. **7**(2), 97–105 (2019)
20. Prabhu, S.S.K.K.: Restaurant recommendation system using machine learning algorithms. International Journal of Science Engineering and Technology. **9**, 3 (2021)
21. Shambour, Q.Y., Abualhaj, M.M., Abu-Shareha, A.A.: Restaurant recommendations based on multi-criteria recommendation algorithm. J. Univ. Comput. Sci. **29**(2), 179 (2023)
22. Hartanto, M., Utama, D.N.: 'Intelligent decision support model for recommending restaurant', Cogent. Engineering. **7**(1), 1763888 (2020)
23. Rahman, M.S., Sarkar, T.D., Emon, M.Y.: Comprehensive geospatial and statistical analysis of restaurant distribution using K-means clustering in machine learning technology. In: Comprehensive Geospatial and Statistical Analysis of Restaurant Distribution Using K-Means Clustering in Machine Learning Technology, pp. 461–466. IEEE (2024)
24. Salih, A.F., Ibrahem, N.F.: Medicine recommendation system based on review sentiment analysis and similarity metrics. In: Medicine Recommendation System Based on Review Sentiment Analysis and Similarity Metrics, pp. 1–7 (IEEE, 2024)