JCTA Journal of Computing Theories and Applications ISSN:3024-9104

Research Article

A Personalized Context-Aware Places of Interest Recommender System

Rianat Abimbola Oguntuase [1], Arome Junior Gabriel [2,*], and Bolanle Adefowoke Ojokoh [3]

[1] Department of Computer Science, Federal Polytechnic Ado-Ekiti, Nigeria; e-mail : bimbson005@yahoo.com
[2] Department of Cybersecurity, The Federal University of Technology, P.M.B. 704, Akure, Nigeria; e-mail : ajgabriel@futa.edu.ng
[3] Department of Information Systems, The Federal University of Technology, P.M.B. 704, Akure, Nigeria; e-mail : baojokoh@futa.edu.ng
* Corresponding Author : Arome Junior Gabriel

Abstract: This research presents a personalized, context-aware recommender system to suggest Places of Interest (POIs) using a hybrid approach combining Bayesian inference and collaborative filtering. The system explicitly addresses the cold-start problem that new users face and improves recommendation accuracy by considering contextual variables such as user mood, budget, companion, and location. The system collects real-time contextual inputs for new users with no historical data and applies Bayesian inference to generate relevant POI suggestions. As users begin to interact and provide ratings, the system progressively shifts to a collaborative filtering mechanism, leveraging cosine similarity to identify similar users within comparable contexts. The recommender system focuses on three categories of POIs: restaurants, hotels, and landmarks. These locations are retrieved through the Google Maps API, and only mapped locations are considered. The system was implemented on Android devices and evaluated through a user study involving 25 participants from diverse backgrounds, including software developers, IT students, and general users. Evaluation metrics such as normalized Discounted Cumulative Gain (nDCG) and classification accuracy were used to assess recommendation quality. Results demonstrate that the system performs better than traditional methods, with nDCG improvements reaching up to 83 percent. Users reported high satisfaction regarding the recommendations' accuracy, ease of use, and contextual relevance. While the system offers significant improvements, it also has certain limitations. Its dependency on Google Maps data may restrict its scope, and using only four contextual factors limits the system's adaptability to more complex user preferences. Future enhancements could include additional dynamic contexts such as weather, POI popularity, and time-related trends, as well as integrating more advanced models to increase personalization and flexibility in real-world applications.

Keywords: Artificial Intelligence; Context-aware systems; Machine learning; Mobile applications; Recommender system.

Received: February, 10th 2025
Revised: March, 28th 2025
Accepted: March, 29th 2025
Published: April, 13th 2025

1. Introduction

A Recommender System (RS) is a software tool and strategy for making suggestions on items (products or services) that are useful to a user [1]. RS became a separate research area in the 1990s. Around that period, the research community began to consider researching recommendation problems that explicitly depend on the concept of ratings as an approach to capturing user preferences for various items. Although some of the existing recommender systems majorly considered recommending items that are most relevant to individual users, they failed to consider any background or contextual information, such as the user's mood as well as their accompanying companions. The importance of this contextual information has been identified in many disciplines, such as information retrieval, data mining, and mobile computing[2], [3]. Indeed, users' contextual information may affect or change the preference of such a user towards an item. Simply put, a user's interest in a given item might be dependent on their specific context.

This current study decided to develop a recommender system for Places of Interest (POIs) because contextual information is crucial for reasoning about user preferences. In contrast to product or movie recommendations, which depend mainly on user-item interactions, POI recommendations are context-aware. Research has proven that user behavior is impacted by many things, such as mood, budget, companions, and location, among other things [2], [3].

An individual might prefer a quiet café for work when they are alone, but they might want the energy of a lively restaurant when with friends. The same goes for the budget, which could move users from a fancy restaurant to a less posh establishment. Notwithstanding these apparent contextual influences, most prevalent recommendation systems tend to use Collaborative Filtering (CF) or Content-Based Filtering (CBF). They are unable to adapt to dynamic changes in user context. On the contrary, a context-aware method guarantees that the suggestions are compatible with some users, yielding additional appropriate and customized recommendations[4]. In this study, we propose a hybrid model based on Bayesian inference (for new users) and Collaborative Filtering (for existing users) to construct a recommendation system that can adapt by considering the changing context of users.

A user may prefer an eatery to a local restaurant when she is with her children. Companions of the user as one of contextual information can influence where to visit. There is no doubt that context plays a vital role in the users' decision process. More interesting and useful suggestions can be achieved if the user's context is known[2].

Context has been perceived differently in various disciplines. Therefore, there exist many definitions of context. Researchers in [5], perceived context as some information with which the current situation of an entity or user can be described. For example, in POI recommendation, the user's mood is a significant contextual information that should form the basis for suggesting a place [6]. Using contextual information in recommenders began as far back as the research work [7]. Choosing the most splendid places under a given context is an important contribution in modern times for strangers or first-time visitors [6]. Factors such as temporal activities, location preferences, user categorical preferences, and even the popularity of POI can influence users' choice of a POI[8].

Mobile Services have recently made life easier and more effective for users. A challenge of high relevance in the recommender system is the lack of contextual experience. The good news is that one of the distinctive features of mobile devices is location awareness, which is the most suitable contextual information necessary for place recommendations[9]. Mobile users usually have their devices almost everywhere. Thus, adding location awareness to recommender systems gives users a more contextual experience. The ability of mobile applications to sense locations and incorporate some other contexts into a recommender system can also provide a better experience.

The major objective of context-based systems is to offer a given user useful info on a product or service depending on the current context of such a user[10]. This objective matches the fundamental objective of a typical recommender system. However, reviewing most of the existing recommendation systems reveals that users' context was largely overlooked. Even where attempts were made to consider these user context variables, only a few were considered. More often than not, the number of these variables considered is directly proportional to the accuracy of recommendation, hence, the motivation to consider as many user context variables as possible in our current study. Furthermore, additional motivation exists to achieve accurate recommendations with a truly lightweight system. Such a system has very low computational resource demand and therefore, is more suitable for resource-constrained devices or platforms such as mobile phones and/or sensors. Hence, the research question here is whether the integration/incorporation of user-centred context variables in the development of POI recommender systems could improve the end-user's experience of the recommendation system.

In light of all these, this current research developed a lightweight user-centered-context-aware mobile recommender for POIs. The location of users, their budget, companion, mood, and other contextual information are considered to achieve improved personalized recommendations. The rest of the paper is organized as follows: Section 2 reviews related research works. Section 3 discusses the proposed work of LSB techniques applied to various color spaces. Section 4 contains experiment results and discussions. Section 5 contains the conclusion as well as suggestions for further studies.

2. Literature Review

This section presents a review of related literature. First, we will briefly discuss and compare the various Context-Based Recommendation (CBR) approaches to highlight their relevance for POIs. Thereafter, we will highlight some of the most related existing research works.

2.1. Context-Based Recommendation Approaches

Here is a brief discussion about how different recommendation systems work. Recommendation systems work based on two main methods: CF and CBF. With CF, the system figures out which users are similar to a given user and then recommends things based on what the other users like. On the other hand, CBF allows a system to recommend items to a given user based on that particular user's preferences, thus suggesting items that match that user's profile.

It is worth noting that CF and CBF methods sometimes fall short when recommending Points of Interest (POIs). This is because they often overlook the users' contextual factors. Imagine how users' current location, time of day, mood, or even budget can change what the users might want.

To overcome this concern, Context-Aware Recommendation Systems (CARS) evolved. These systems are a bit smarter because they consider those extra contextual factors. CARS can be broken down into three main approaches:

1. Pre-filtering Approaches: User context is considered before a recommendation is made.
2. Post-filtering Approaches: In this case, recommendations are made and then adjusted based on user context.
3. Model-based Approaches: Machine learning models aid the recommendation procedure, often leading to more personalized recommendations.

When it comes to recommending points of interest, being aware of the context makes a huge difference. People do not just pick places randomly; they often choose based on what is happening around them. For instance, users may want to go for outdoor spots on a sunny day. Or let us say someone might pick a restaurant based not just on the food, but on who they are hanging out with at the time.

This study takes a hybrid model approach. For new users, we're leveraging Bayesian inference, a fancy way of saying we use probabilities to determine what they might like. For the existing users, CF is used, which helps us make recommendations based on what similar users have enjoyed. This way, recommendations are personalized and adaptable to changing circumstances. This ultimately ensures that recommendations feel right for each individual, depending on the context they find themselves in.

2.2. Related Works

Numerous related research works exist in the literature. This section highlights some of the most related ones.

The paper in [11] argued that there is an additional benefit of context management beyond improving the recommendation accuracy of the core recommendation algorithm. They developed a system that considers the active user's contextual situation to explain why an item has been recommended. The work was motivated by the need for a system that can provide the reason(s) why a given item gets recommended. The system was named ReRex. Their results indicate that context management and explanations provided with each contextual situation improve user satisfaction with the recommender system. The time dimension, such as the day of the week, was not incorporated.

The study in [12] presented a fuzzy logic-based system that mines information about laptops and uses this information in a formal model to suggest items to target users. The system recommends laptops to prospective buyers, supplying them with information on the products that could best satisfy their needs. Their work was motivated by the need to develop a platform that aided customers and boosted the sellers' sales value. The system was not implemented on mobile platforms, and contextual information was not considered.

In [13], a context-based mobile recommender for POIs that leverages weather-context information for making highly relevant suggestions to users was presented. Their proposed system was implemented as an always-on rich client design. The Android client is made of a graphical user interface and a component for the presentation logic; the server serves as the repository for the whole recommendation logic and even the data layer. The server uses web services or data repositories made available by the Regional Association of South Tyrol's Tourism Organizations, the Municipality of Bolzano and Mondometeo, which allows for the acquisition of graphical or textual descriptions and even weather-related information for about 27,000 POIs. The users specify the time, which is automatically detected by the system.

The article in [14], presented a web-based recommender system for suggesting mobile phones to the target prospective buyers. A decision support tool was used to consider some attributes of mobile devices, and they were analyzed based on customers' preferences and classified into different hierarchical levels. The work was motivated by the necessity of an interactive computer-based approach that provides logical comparisons and regular/reliable mobile product recommendations. Though this system recommends mobile devices to users, it is web-based and cannot run on mobile platforms.

Liu [15] proposed the use of context history in mobile recommenders. A system the authors named CAMTRS, which has four major modules, was developed based on this. The systems' modules include context history acquisition, inference, preference prediction, user modeling, and the actual recommendation module. The work was motivated by most Recommender Systems considering sanctioning content-based techniques on user preferences. Nevertheless, user inclinations may differ in diverse contexts, such as at different locations or times of day. Consequently, there is a need to introduce actual user context into Recommender Systems. Also, several research works ignore context history while considering only the user's current context. The model in Liu's research article adopted the CBF method for recommending relevant tourist info to users. The system adopts the form-filling method to obtain users' contexts.

The research [16] presented a Bi-Objective Recommendation Framework (BORF), a hybrid cloud-oriented mobile social grid. Based on this proposed framework, a system called MobiContext was presented. The authors used sparse Geo-Social Networking Data to develop a Location Based Preference Aware Recommender. This was motivated by most location-aware preference recommendation systems considering only user-related limitations as filtering conditions when planning a journey. The authors opined that. According to the work, most previous work on collaborative filtering-oriented recommendation suffers from challenges such as cold start, scalability, and even data sparseness. Also, the recommendation problem often comes with the existence of some decision variables or conflicting goals, such as users' preferences and venue closeness. Hence, the MobiContext generated personalized suggestions or recommendations using multi-goal optimization methods. BORF uses the Hub-AVERAGE (HA) deduction model to address the challenges that bother data sparseness and cold start. Contextual information, such as the goal of the visit and the weather, was not considered.

The study in [8] presented a preference-aware POI recommendation system. The system suggests to users that places within a geo-spatial range match their temporal activities and categorical preferences. The work combined four factors (spatial, temporal, categorical preferences, and POI popularity) to generate recommendations. In [6], a context-based recommender was presented, which suggests places to users by considering factors such as the current mood, weather, and the time of the day of the user. This system automatically detects the weather and time of day around a user's location. Combining these factors as well as the current mood of the user, the system then chooses the location where the user should go. Places are suggested to users based on previous locations/places such users have visited under similar contexts or conditions. The challenges to generating CBR are known by detecting the related contextual factors like the weather and/or user's mood, which affect the ratings and are worth examining. A genetic algorithm was used to predict ratings for each place in the mobile recommender application they developed. The authors named their "Context-Aware Genetic Recommender System" (CAGRS). In [3], a context-based tourism recommender, which solves the problem of information overload by considering users' preferences when making personalized recommendations, was presented. Contextual info was introduced to allow for improved recommendations. Their proposed system used an ontology approach to overcome the problem of the absence of a formal definition for representing contextual information and user's preferences, which was present in previous context-conscious tourism recommenders. Their proposed system learns the user profile dynamically using their feedback. The authors proposed a method that allocates more effect in the spreading procedure for nodes. Personalized and contextualized recommendations suggest the most suitable places of interest near the user's location. The user's mood was not also considered as one of the contextual factors.

Kuanr et al. [17] developed a CF technique-based recommender for use in the tourism domain of human life. This was achieved by grouping users according to their choice of place, food, and even local item purchases. Using data acquired about these items, together with the ratings past users have given about these items' providers or sellers, the proposed system can make recommendations of the best spots (shops), along with good food and products available in those spots, to a new user or tourist. The results supported the claim that CF is the more reliable technique for personalized recommender systems. However, increasing the number of users' context variables considered will improve the recommendations.

Similarly, the need to overcome most recommender systems' high computational resource demand and privacy concerns motivated the research [18]. The research proposed a novel Light Location Recommender System (LLRec) to perform the next POI recommendation locally on resource-constrained mobile devices to achieve this. FastGRNN, a lightweight but effective gated Recurrent Neural Network (RNN), was adopted. Their approach significantly reduced the dependency of recommendations on cloud servers, thus allowing for the next POI recommendation in a stable, cost-effective, and secure way.

The study in [19] considered recommendation services in the agricultural or food production domain. This study designed a collaborative system that can offer farmers recommendations on crops to cultivate or plant depending on contextual variables like the farmer's location, and even the weather conditions of the previous months. The system developed can also recommend other seeds, pesticides, and farm implements based on the farmer's context or preferences. Other measures adopted in this work include the cosine similarity measure and fuzzy logic. The results reveal that it provides a prior idea regarding a crop before sowing seeds. However, increasing the number of users' context variables considered will improve the recommendations.

The research in [20] carried out a study on developing a recommender system using item-based collaborative filtering and K-Means techniques. Context variables considered in this study include the users' attentiveness, taste, and unique behavior. The work is limited to m users and n items (in numbers). The proposed system presented a model for recommending n number of items to m number of users. Considering more user contextual-variable values will be a way to improve their work.

Zhang and Wang [21] proposed a medical-oriented recommendation system in which patient's background data is used to bootstrap the collaborative filtering engine and personalized suggestions are provided therein. Results demonstrated that the content-bootstrapped part of the system enhanced the effectiveness of the CF recommendation in medical articles.

Most existing research has honed in on CF and CBF. These methods work pretty well when considering static preferences. However, they come short when it comes to dynamically changing contexts. That is where this research steps in. There is a real need to fill that gap. The goal is to develop a hybrid approach to adjust to how user contexts evolve. We hybridize Bayesian inference for new users and Collaborative Filtering for existing users to achieve this. This could boost how personalized and adaptable POI recommendations can be. This approach keeps those recommendations fresh and relevant, even when users' preferences change, whether because of changes in their moods, financial situations getting tight, or social influences coming into play. It must also be said that in contrast to some existing related studies, this current research considered more than one contextual factor.

3. Proposed Method

This section discusses the overall design of the proposed system. The section begins with presenting our proposed system's architecture in Figure 1. The system architecture shown in Figure 1 reveals that the system has seven main components/parts. The function of each system part is presented as follows:

3.1. Registration

This component handles the registration of users by prompting the entry of their personal information, such as username, e-mail address, password, and profession, into the system. A user cannot access the recommender until after completing the registration procedure.

Once the registration process is completed, the page where the user can log in is displayed, prompting the user to enter a username and password to use the system.

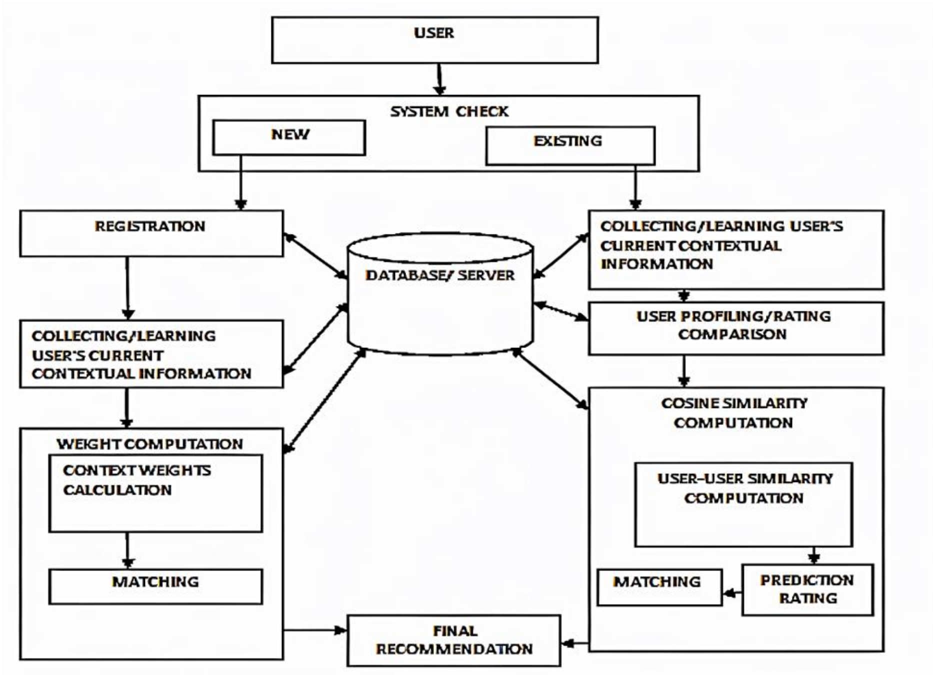


Figure 1: Architecture of the Proposed System

3.2. Collection of Context Data and Calculation

Context refers to a narrative of the circumstances and the environment/setting of the user or device under consideration [22]. A set of features is relevant for each context. A series of values is deduced for each feature by the context. The set of contextual information used in the proposed system, with their respective values and whether they are explicitly or implicitly obtained, are presented as follows:

- Location – distance_to_POI (implicit): very_near, near, far, very_far;
- User's mood (explicit): happy, tired, hungry, bored, adventurous;
- Budget (explicit): less_than_N1000, Btw N1000_and_N5000, Greater_than_N5000;
- Companion (explicit): alone, children, family, spouse, and friends.

The weights assigned to each context in the database are calculated to give recommendations for new users. Users supply the user's mood, budget, and companion. At the same time, distance_ to_POI is detected automatically by the system, and each is described along with options as presented in the database, which is shown in Table 1.

Table 1. An explanation of the context data and the drop-down options

S/N	Context	Options
1	User's Mood	Happy, Tired, Bored, Hungry, Adventurous
2	Budget	<N1000, N1000<Budget<N5000, >N5000
3	Companions	Alone, Children, Family, Spouse, Friends
4	Distance_to_POI	Very_Near, Near, Far, Very_Far

The proposed system relies on four main factors: mood, budget, companion, and location. These four factors are straightforward and give us the information to help people decide where to go. The users' moods and how much money they have can change in a flash, and those shifts can affect their choices. Additionally, who the users are with matters, especially when considering where to eat or travel. Other factors like the weather, time of day, how popular a place is, or even the season, which could make better recommendations, can also come here. The challenge, however, will be that the level of complexity in the recommender system will significantly increase. This ultimately will lead to increased computational complexity or resource demand by the proposed system. Therefore, even though there is potential to explore those extra factors to make recommendations even more accurate, this study focuses on those four key elements (mood, budget, companion, and location). These are the ones that really matter and can be measured easily. This way, we can keep everything practical and efficient, thus balancing the proposed system's performance with simplicity.

POI distance that is less than two kilometres is assigned variable "Very_Near", distance that is between two kilometres and five kilometres is assigned "Near", the one that is between five kilometres and ten kilometres is "Far" and distance more than ten kilometres is "Very_Far".

POIs are classified based on the user's mood since a user's mood has a major role to play in the choice of a place of interest, according to experts (Psychologists). The weight associated with a budget, companion, and Distance_to_POI is 0.5, 0.2, and 0.3, respectively. The weights allocated to each of these contexts in the database are used for calculations using the Bayesian algorithm, and POIs in the range of values are suggested for the new user as presented in Equation (1).

$$P(U) = w_1 * v_w + w_2 * v_w + w_3 * v_w \tag{1}$$

where w_1, w_2, and w_3 are the weight of each context and vw is the corresponding variable weight of the context.

The system calculates a preference score $P(POI)$ for each Point of Interest based on the contextual variables. These scores are then compared to the active user's contextual preference score $P(U)$. Only POIs with scores satisfying the condition: $P(U) - 0.05 \leq P(POI) \leq P(U) + 0.05$, are considered contextually relevant and are recommended to the user.

Let user mood M, be given by the relation in Equation (2).

$$M = \{m_1, m_2, m_3, m_4, m_5\} \tag{2}$$

Also, let $POI\ P$, be represented by the relation in Equation (3).

$$P = \{p_1, p_2, p_3\} \tag{3}$$

where m_1, m_2, m_3, m_4, m_5 represent hungry, happy, adventurous, tired, and bored, respectively. Meanwhile p_1, p_2, p_3 denote restaurant, hotel, and landmark, respectively. As an example of mood-to-POI associations is shown in Equation (4).

$$\begin{gathered} m_1 \rightarrow p_1; \\ m_2 \rightarrow \{p_2, p_3\}; \\ m_3 \rightarrow p_3; \\ m_4 \rightarrow p_4; \\ m_5 \rightarrow p_5; \end{gathered} \tag{4}$$

Some of the possible POIs with associated user's mood as represented in the database is presented in Table 2.

The categorized POIs in the system's database are as follows:

- Hotels: Rejov Hotel (Aisegba Ekiti), Koltotel Plaza and Suites, AB Emporium Hotel and Event Centre, Dave Hotel, Midas Hotels Limited, Radjut Hotel, Olart Hotel, and Suites, Great Expectation Hotel and Resort, College of Education Hotel (Ikere Ekiti), De Xambdra International Hotel, Highflyer Hotels and Resort, Home Away Hotel, Midas Hotel and Arena, Park View Hotel, Pathfinder Hotel and Holiday Inn, De Jewels Apartments and Suites, Prosperous Royal Hotel and Resort, Queen's Court Hotel, Yemraf Hotel (Ado Ekiti), South Western Hotels Ltd.
- Restaurants: Chicken Republic, Portofino Eateries, Tasty and Spicy Restaurant, Captain Cook Eatery, Take Away Eatery, Danke Fast Foods, Meet Me Der Eatery, Ewa Wa Kitchen, TOZ Bakery, Tantalizers (Ado Ekiti).

- Landmarks: Orole Hill (Ikere Ekiti), Olosunta Hill (Ikere Ekiti), Agbonna Hill (Oke-Mesi), Motif Glitz Ventures / Motif Funland (Iworoko Ekiti), Adekunle Fajuyi Memorial Park (Ado Ekiti), Arinta Waterfall (Ipole-Iloro), Erin Ijesha Waterfall, Ikogosi Warm Spring (Ikogosi Ekiti), Esa Cave (Iyin Ekiti), EKSU Torch Museum (Ekiti State University), Ewi's Palace (Ado Ekiti), Ologotun's Palace, Ekiti State Golf Club.

Table 2. Some appropriate POIs for each users' mood (new user).

POI	Mood
Koltotel Plaza & Suites, AB Emporium Hotel and Event Centre, Dave Hotel, Motif Glitz ventures (Motif Funland), Adekunle Fajuyi memorial park	Happy
Midas Hotels Limited, Radjut Hotel, Olart Hotel and Suites, Great Expectation Hotel & Resort, College of Education hotel	Tired
Arinta Waterfall, Ekiti State Golf Club, Ikogosi Warm Spring, Erin Ijesha Waterfall,	Bored
Chicken Republic, Portofino eateries, Tasty and Spicy Restaurant, Captain Cook Eatery, Take Away Eatery.	Hungry
Motif Glitz ventures (Motif Funland), Adekunle Fajuyi Memorial Park, Arinta Waterfall, Erin Ijesha Waterfall, Ikogosi Warm Spring	Adventurous

The location of the user needs, the direction, and the distance to these POIs are achieved using Google Map API.

3.3. User Profiling/Rate Comparison

At entry, the system offers a registration/ login page. This is to keep a log of users' data in the database, which will be used as input along with the available context data in the collaborative part of the proposed system. The user is requested to rate a series of POIs after each recommendation, which helps the system provide better subsequent recommendations [13]. The rating data varies from 1 to 5 stars. The user's response is saved against the context values he rated the system. These data are fetched and used for calculation when similar users request recommendations in similar contexts. The system automatically uses the collaborative part to make recommendations if a user exists. The system extracts necessary information and ratings that the user previously supplied.

3.4. Collaborative Filtering (CF): Cosine Similarity Computation

This current work adopted the user-based CF technique. The cosine similarity method was used to calculate the similarity between the users' ratings. Cosine Similarity calculates the similarity between two users by finding the cosine of the angle between the two vectors, as presented in Equation (5).

$$\text{sim}(a,u) = \cos(\vec{a},\vec{u}) = \frac{\vec{a}\cdot\vec{u}}{|\vec{a}|\cdot|\vec{u}|} = \frac{\sum_{i=1}^{k} a_i u_i}{\sqrt{\sum_{i=1}^{k}(a_i)^2}\cdot\sqrt{\sum_{i=1}^{k}(u_i)^2}} \tag{5}$$

where a_i and u_i represent the ratings given by the active user a and a neighboring user u for item i. The similarity score ranges from -1 (perfectly dissimilar) to 1 (perfectly similar). To predict the rating of the active user a for a particular item i, a weighted average of the ratings from similar users is computed using Equation (6).

$$p_{a,i} = \bar{\bar{r}}_a + \frac{\sum_{u=1}^{n}\text{sim}(a,u)\cdot\left(r_{u,i}-\bar{r}_u\right)}{\sum_{u=1}^{n}|\text{sim}(a,u)|} \tag{6}$$

Where $\bar{\bar{r}}_a$ and $\bar{r}_u$ are the average ratings of users a and u, respectively.

3.5. Context-Aware CF System

For each context type, defining a similarity measure between different context values is essential. Context is represented as a tuple of n different types:

$$C = \{c_1, c_2, c_3, \cdots, c_n\} \tag{7}$$

where c_t(with$t \in \{1, \cdots, n\}$) represents a context type such as user mood, budget, or companion. For each context type t, a similarity function $\text{sim}_t\ (x, y)$ is defined, where $x, y \in C$, and returns a normalized value indicating the similarity between x and y with respect to c_t.

Cosine similarity, as defined in Equation (5), is also used to compute similarity between context values. Let $r_{u,i,x}$ be the rating by user u on item i under context x. Then, the similarity between two context values x and y for item i is defined in Equation (8),

$$\text{sim}_t(x, y, i) = \frac{\sum_{j=1}^{k} u_{j,x_t} u_{j,y_t}}{\sqrt{\sum_{i=1}^{k} u_{j,x_t}^2} \cdot \sqrt{\sum_{i=1}^{k} u_{j,y_t}^2}} \tag{8}$$

Each rating is associated with a context. To determine how relevant a rating is in the active user's current context c, we define the context-weighted rating $r_{u,i,c}$ using Equation (9).

$$r_{u,i,c} = k \sum_{x \in C} \sum_{t=1}^{z} r_{u,i,x} \cdot \text{sim}_t(c, x) \tag{9}$$

Where z is the number of context dimensions, and k is a normalizing factor ensuring the sum of the weights to 1. The summation iterates over all dimensions (e.g., budget) and their values (e.g., "less than N1000", "between N1000 and N5000", "greater than N5000").

The normalization factor k is given by:

$$k = \frac{1}{\sum_{u=1}^{n} |\text{sim}(a, u)|} \tag{10}$$

Substituting $r_{u,i,c}$ into Equation (6) allows us to compute the predicted rating $p_{a,i,c}$ for the active user a, item i, and context c, as shown in Equation (11).

$$p_{a,i,c} = \bar{\bar{r}}_a + \frac{\sum_{u=1}^{n} \text{sim}(a, u) \cdot (r_{u,i,c} - \bar{r}_u)}{\sum_{u=1}^{n} |\text{sim}(a, u)|} \tag{11}$$

This formulation ensures that the predicted rating incorporates both user similarity and contextual relevance.

3.6. Database Design

The database serves as the backend system which provides supporting services. It has embedded heuristics rules. It is also an avenue where information collated from the user, such as profile data and context data, will be stored, as well as software data obtained from learning users' behavior, amongst other data. It will generally hold a chunk of data on which recommendations are based.

3.7. Final Recommendation

The output of the proposed system consists of a list of places recommended as "FINAL RECOMMENDATION" for users per the weights of contextual information supplied for the new user and the choice of similar previous users in similar contexts. Recommendations are obtained from the results of each subsystem and displayed at the interface with pictures of the POIs along with map directions/routes to such POIs for the users' use.'

3.8. Addressing Differences in Methods for New and Existing Users

3.8.1. The Cold-Start Problem and Transition to Collaborative Filtering

The cold-start problem is one of the main challenges in recommender systems, which arises when a brand-new user does not have enough interaction data (ratings or view history) to generate reliable recommendations using CF. Since CF is based on user-item interactions and user preference, a new user who has made no ratings nor been engaged with any items cannot receive recommendations based on CF right away.

To solve this problem, we use Bayesian inference, which considers several contextual factors (mood, budget, location, and companions) to make recommendations, even before the user rates any POIs. As a user interacts with the system and gives feedback over time, a threshold is passed to switch to CF.

3.8.2. Comparison of Bayesian Inference vs. Collaborative Filtering

Table 3. Differences in Recommendation Strategies for New vs. Existing Users.

Feature	Bayesian Inference (New Users)	Collaborative Filtering (Existing Users)
Dependency on Prior Ratings	There is no need for prior ratings	Prior user ratings are needed
Adaptability to Context	Directly uses context factors (mood, budget, etc.)	User contexts or preferences are learned over time
Personalization	This is based on general trends. It is therefore limited.	This is based on users' choice or preferences
Computational Complexity	Little	This is usually higher owing to the need for similarity calculations
Transition Mechanism	Here, context-based POI selection	Switches when the user has enough ratings

3.8.3 How Recommendations Improve Over Time

When this model becomes able to learn by receiving feedback whenever a visitor gives a rating, it creates a recommendation model that improves a recommendation engine based on user ratings. The more users give ratings, the more data the system is trained on. There are stages through which the system evolves;

- Phase 1: New User (Bayesian Inference) – Users are shown Point of Interest (POI) suggestions based on static contextual attributes.
- Phase 2–Initial Ratings and Feedback: The user indicates preliminary ratings, and the system begins recording the user's preferences.
- Phase 3-Transition Period — The system integrates Collaborative Filtering while the user proceeds along this activity trajectory toward regulated ratings.
- Phase 4-User Development Stage –The user avails of fully customized recommendations.

The fact that we structured the recommendation procedure this way enables the system to be helpful to both new and existing users. This will, by extension, eliminate or reduce the cold-start issue and go a long way toward producing better recommendations as more user data becomes available.

3.9. Our Proposed Approach versus Complex AI/ML Approaches

The approach proposed in this current study for the recommendation of POIs, especially on mobile platforms, offers some more advantages or benefits over complex methods like Deep Learning (DL) and Reinforcement Learning (RL);

4. Results and Discussion

The proposed system is deployed on Android devices. Experiments were conducted on an environment with a Windows 10 Operating System platform running on an Intel Pentium (R) Processor and 4GB Random Access Memory.

4.1. Proposed Recommender User Interfaces

Typically, interaction with the proposed POI recommendation system involves several phases. First of all, the user will establish their "context". This is done via the system Graphical User Interface (GUI). Each user selects or sets context variable values representing important contextual factors. The users may also ignore setting these contextual variables if they so wish, in which case, the eventual recommendation would be a bit non-specific or somewhat more general. Some of these context variables relate to the users' mood, budget, or number of companions. When these context variables are selected, the recommendation system adjusts accordingly to accommodate the users' selections in the recommendation procedure. While some of these context values (such as the user's location, distances of POIs from the user's current location, etc.) can be obtained automatically by the recommender system, the user is expected to make selections in some other cases (such as user's mood, budget or companions).

The full set of contextual variables considered in this research, their values, and whether they are automatically collected or not is provided as follows (and as presented in Table 1):

- Distance to the POI (automatic/implicit): Very_Near, Near, Far, Very_Near;
- Companion (user selection): alone, friends, family, partner, children;
- Mood (user selection): hungry, happy, tired, bored, adventurous;
- Budget (user selection): less_than_N1000, Btw_N1000_and_N5000, Greater_than_N5000;

Once the user selects or specifies the contextual situation, the recommendation system provides the required recommendations. The proposed system is a client/server application. All recommendation requests are sent to a server designated to store the user ratings and the current request context and compute the recommendations.

At the point of entry, the system offers a registration page that allows the user to enter his username, e-mail, password, and profession. The user can log into the system afterward with the username and password supplied at the registration phase. Then, some forms are displayed, allowing the user to select options that depict his present context and relate to his mood, budget, and companion. The system implicitly learns other contextual information, such as location (distance_to_POI). The user's mood has options such as happy, tired, bored, hungry, and adventurous. Budget possesses options such as less_ than_N1000, Btw N1000_and_ N5000, and Greater_than_N5000. Companion has options such as Alone, Children, Family, Spouse, and Friends. On the other hand, distance_to_POI is learned implicitly.

In this sub-section, some of the interfaces of our proposed recommender system, through which any user can interact with the system to get desired recommendations, are presented. Figure 2, for instance, shows the context data supplied by the active user on the recommender system/ application. In contrast, Figure 3 presents information about the places recommended by our system to the active user.

4.2. Proposed System Evaluation

To evaluate our proposed system, the questionnaire approach was adopted. Here, well-structured questionnaires were administered to obtain responses from users of our developed system. The context-aware recommender system was presented to a group of 25 users (that is, three system developers, six IT-inclined students from the School of Science and Computer Studies at the Federal Polytechnic, Ado-Ekiti, Nigeria, and sixteen other users) to evaluate the system with the aid of questionnaires. The questionnaires administered contained questions from 1 to 12, and five-point scaled variables: Strongly Agree, Agree, Fairly Agree, Indifferent, and Don't Agree were used. Questions were asked about the accuracy, the importance of context in the recommendation, friendliness to the user, correctness, understandability, and acceptability of the system. Results obtained are plotted in graphs as shown in Figures 4 and 5. The analysis of the response is shown in Figure 4.

We did a thorough questionnaire survey to appraise how usable and effective the system is. We administered the questionnaire to a varied group of 25 people. Among them were three system developers, six students who are into IT from the School of Science and Computer Studies at the Federal Polytechnic in Ado-Ekiti, Nigeria, and sixteen everyday users from the local area. Talking about the demographics, the respondents' ages ranged from 18 to 55. About 60% were male and 40% female. As for education, we had people from undergrad to postgrad levels.

Additionally, our sample population included both locals from Ekiti state and some visitors who were just passing through. The idea behind this diversity was to capture a wide range of user experiences. We wanted to make sure that when we evaluated the recommender system, it truly reflected how it would work in different real-life situations.

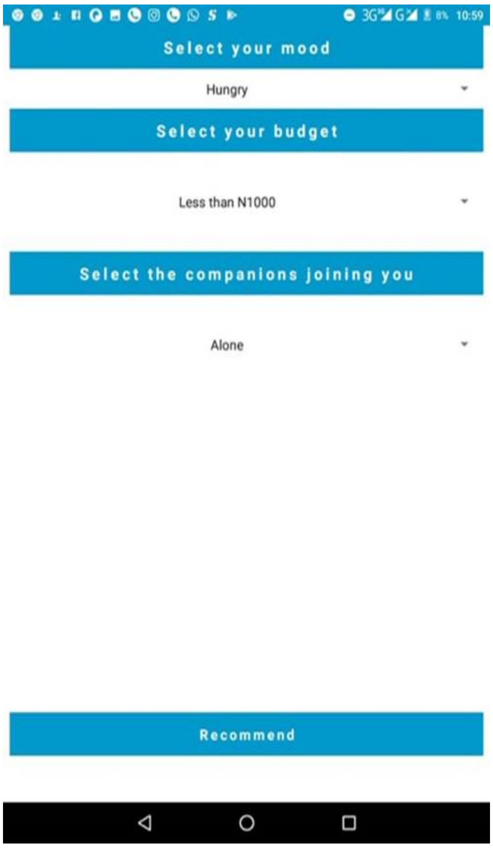


Figure 2. The User Context Interface

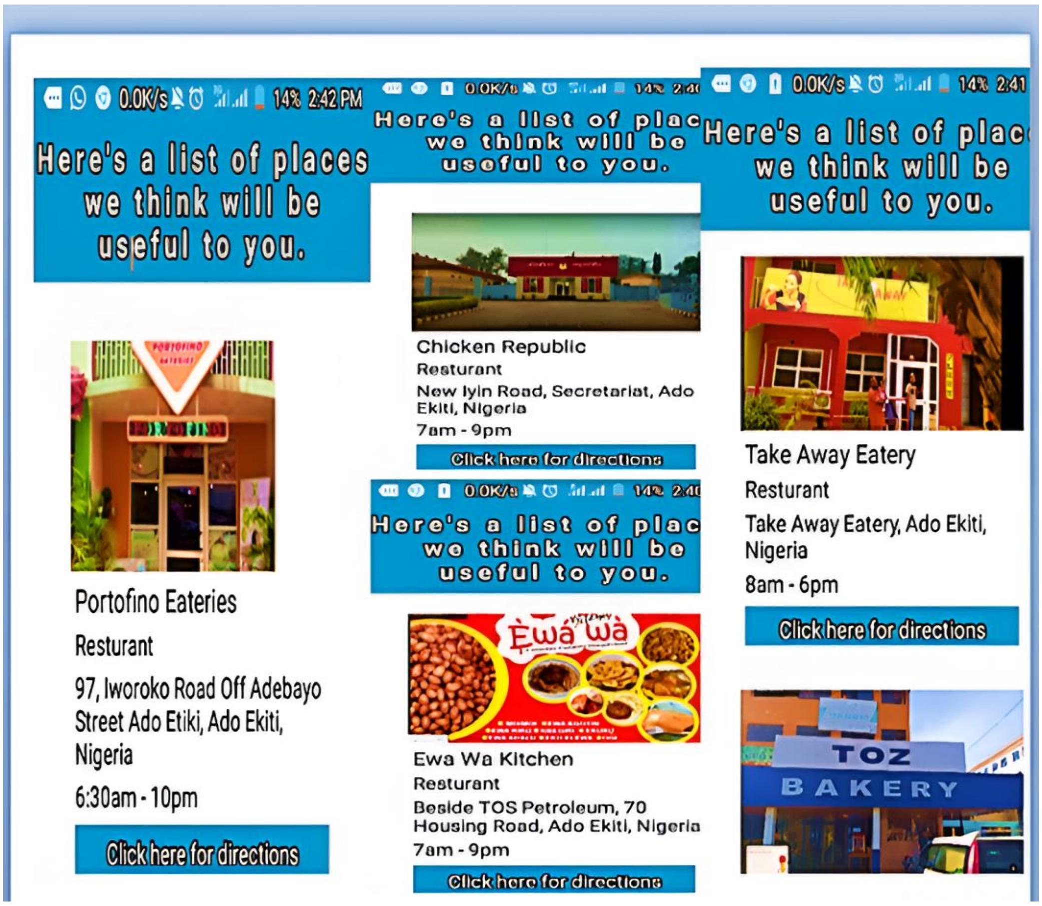
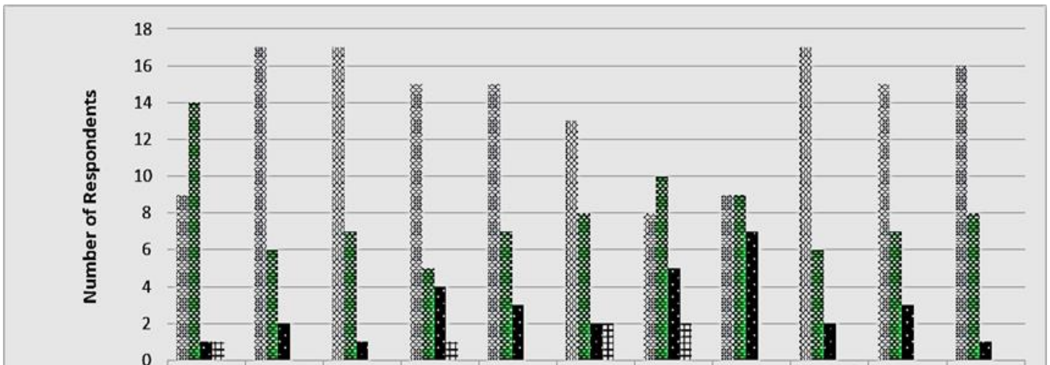


Figure 3. Some of the recommended POIs.



	Recommendation matched my Interest	Flexible	User–friendly	Acceptable Speed	Easy to use	Accurate	POIs are ranked correctly	Satisfactory	Context is essential in recommenders	Recommend System to Other users	Will use system again
Strongly Agree	9	17	17	15	15	13	8	9	17	15	16
Agree	14	6	7	5	7	8	10	9	6	7	8
Fairly Agree	1	2	1	4	3	2	5	7	2	3	1
Indifferent	1	0	0	1	0	2	2	0	0	0	0
Disagree	0	0	0	0	0	0	0	0	0	0	0

Figure 4. Users' Rating of the Proposed System.

Figure 4 shows that 40% of the users strongly agreed that the places they recommended matched their interests, 52% agreed, 4% agreed, and 4% were indifferent. This shows that almost all the users expressed satisfaction with the recommendations. Concerning how easy (or not) it was for the users to modify their tastes based on their mood, budget, and companion, about 68% of the users strongly agreed to enjoy ease and comfort while using the system, 24% agreed, 8% fairly agreed and this shows that the system considered personalization of recommendation. Regarding the user-friendliness of the developed system's interfaces, 68% of our respondents strongly agreed that the interface is friendly, 28% agreed, and 4% fairly agreed. This implies that the user interface(s) exhibit a high level of user-friendliness. With regards to the speed of recommendation, 60% of the users strongly agreed that the system gave recommendations with a decent speed, 20% agreed, 16% fairly agreed while 4% were indifferent about the speed of recommendation, as what matters to them could probably be the accuracy of such recommendations and not necessarily the speed. On whether the comprehensiveness of our POI recommender system is high enough or adequate, 60% of the population sample strongly agreed that the system is not difficult to use and understand, 28% agreed, while only 12% fairly agreed. 52% strongly agreed that the system gave accurate recommendations. As per the correctness of the POI ranking order (according to users' preferences), 32% of our respondents strongly agreed that places were ranked correctly according to their preference, 40% agreed, 20% fairly agreed, and 8% were indifferent, which shows that the order in which places are recommended were ranked correctly. Similarly, when referring to the perception of the sample population on the overall performance of our recommender system, 36% strongly agreed that the system performance is adequate, 36% agreed, and 28% fairly agreed.

Regarding the significance of users' context in POI recommendations, 68% of respondents strongly agreed that user context is an important ingredient in place recommendations, 24% agreed, and 8% fairly agreed. Considering users' perception of their continuous or repeated usage of the system for obtaining POI recommendations, 60% of the users strongly agreed to re-use the system, 28% agreed, and 12% fairly agreed. Also, 64% of the sample population strongly agreed that the system can be useful for the general public, 32% agreed, and 4% fairly agreed.

Summarily, it is clear from the analysis of the responses obtained from the users that this POI recommendation system performs excellently well as it provides recommendations according to the preferred choice of the users, such that a larger percentage would love to use the system again and also recommend it to other people. Our proposed system generally meets users' experience goals, especially regarding accuracy, ease of use, user-friendliness, speed, correctness of ranking, and even flexibility. Almost all the users agreed as in shown in Figures 4 and 5, to the fact that context variables are crucial to the accuracy of personalized recommendations. Discounted cumulative gain (DCG) is used for measuring ranking quality.

4.2.1 Evaluation of Ranking Quality

In this research, the recommendation process is treated as a ranking task. This means that only a limited number of the most relevant items are presented to the user at any given time. Such an approach is commonly referred to as a top-K recommendation. Among the widely used ranking evaluation metrics, Mean Average Precision (MAP) and Discounted Cumulative Gain (DCG) are the most prominent. In this study, DCG is adopted to evaluate the ranking quality.

Discounted Cumulative Gain (DCG) measures how well the recommended items are ordered. The rationale for using DCG in recommender systems lies in assessing the usefulness of the ranked list returned by the system. Each item in the recommendation list is assigned a relevance score, usually a non-negative value, referred to as "gain". For items without user feedback, the gain is assumed to be zero. DCG accumulates the gain from the top of the ranked list, with the gain values of items in lower positions being reduced (discounted) by a logarithmic factor.

Since DCG values can vary between users, normalization is necessary for fair comparison. The worst possible DCG value is zero. To obtain a normalized score, the ideal DCG (IDCG) is calculated by ranking the items in the most optimal order (based on relevance), and computing the DCG of this ideal list. The normalized DCG (nDCG) is then obtained by dividing the actual DCG by the IDCG. This yields a value between 0 and 1. The formulas are given in Equations (12), (13), and (14).

$$\mathrm{DCG}_p = \mathrm{rel}_1 + \sum_{i=2}^{p} \frac{\mathrm{rel}_i}{\log_2(i)} \tag{12}$$

where p denotes the position up to which relevance is accumulated and rel_i is the relevance of the item at position i.

$$\mathrm{IDCG}_p = \mathrm{rel}_1 + \sum_{i=2}^{|h|-1} \frac{\mathrm{rel}_i}{\log_2(i)} \tag{13}$$

$$nDCG_p = \frac{\mathrm{DCG}_p}{\mathrm{IDCG}_p} \qquad (13)$$

Table 4 presents an example of a user's recommendation result, showing the original item rankings in the database.

4.2.2. Test of Reliability and Validity of the Questionnaire

Additionally, the system evaluation questionnaire needs to be reliable and valid. Therefore, the following steps were taken:

- Reliability Testing Analysis: The internal consistency of the questionnaire items was assessed using Cronbach's Alpha. The result was greater than 0.7, indicating good reliability and consistent responses across the items.
- Validity Testing: Content and construct validity were established by consulting domain experts, who conducted a pre-test with a small group of respondents before final deployment. This process ensured that the questionnaire items were relevant, clear, and able to measure key constructs such as user experience and system performance.

These steps contributed to the credibility of the collected data and supported the validity of the results obtained from user testing. Figure 5 presents the comparative analysis between two existing related works and the proposed POI recommender system. The study in [23] uses a content-based approach to food recommendations. In contrast, the study in [24] recommends places similar to the present work but does not consider user mood as a contextual factor. The average accuracy and efficiency scores obtained from the comparative analysis are 76%, 80%, and 83% for [23], [24], and the proposed system, respectively. These results indicate that the proposed system delivers more accurate and efficient recommendations. The scores shown in Figure 5 are derived from a combination of normalized Discounted

Cumulative Gain (nDCG) and classification accuracy, as previously defined in Equations (12) to (14). These metrics provide a comprehensive evaluation of both the ranking quality and the recommendations' alignment with user preferences.

Table 4. Order of Items in the Database versus the Proposed System's Recommendation.

Order as in the Database	Order as Recommended by the Proposed System
Portofino Eateries	Chicken Republic
Meet me Der Eatery	Ewa wa Kitchen
Take away Eatery	Meet me Der Eatery
Chicken Republic	Take away Eatery
Ewa wa Kitchen	Portofino Eateries

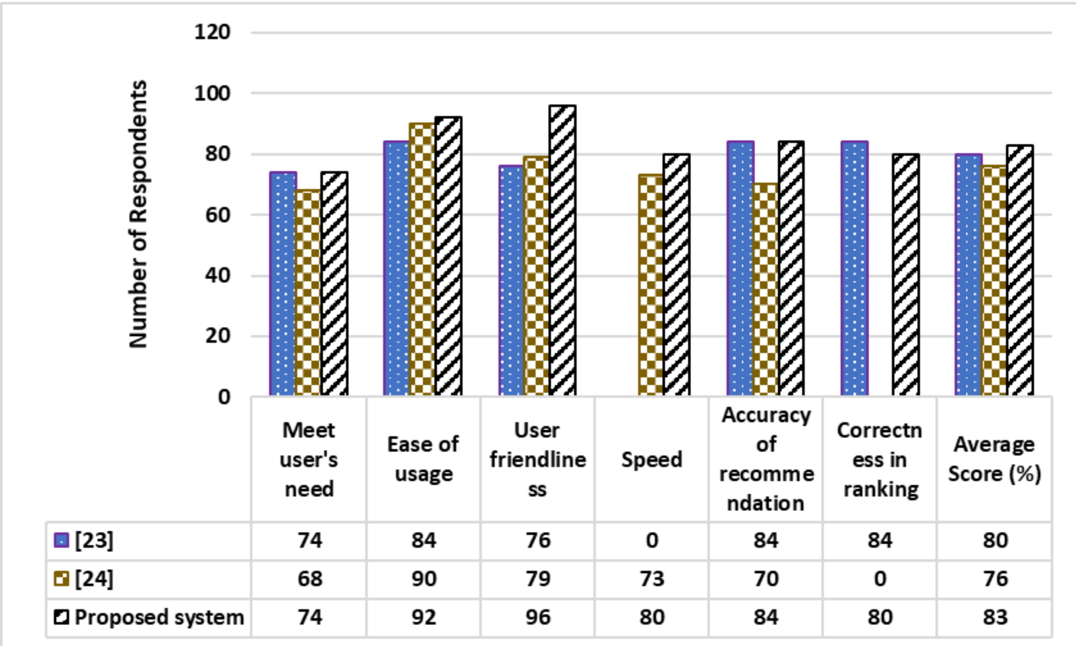


Figure 5. Comparison between the Proposed System and Existing Related Systems

4.2.3 Ablation Study Comparing nDCG results with different filtering approaches.

So, to back up how effective our hybrid method is, we carried out an ablation study. This study compared how well our Context-Aware Collaborative Filtering (CACF) model performed against the two base methods. First up, we have the traditional CF, which does not consider context but just looks at user-item interactions. Then there is CBF, a method that focuses on recommending POIs based purely on how similar the items are in terms of their descriptive features.

To appraise the performance of these methods, we used the nDCG. This metric helps us figure out the quality of the recommendation rankings. Our results revealed that the CACF model scored an impressive average nDCG of 83%, the traditional CF model, which ignored context, came in at 80%, and the CBF method ended up at 76%. This clearly shows that when one adds contextual information to the recommendation process, the ranking quality gets boosted. The improvement from 80% with the CF model to 83% with CACF highlights how valuable context-aware approaches can be. Additionally, that lower score from the CBF method just drives home the point about how important it is to consider user behavior data alongside context

5. Conclusion and Future Works

This research paper presented a recommender that hybridizes the two major techniques of the recommendation process (the context-aware technique and the collaborative filtering approach), to recommend or suggest places of interest to users effectively. The weights of contexts are captured and used to calculate the most suitable places of interest for the new user using the Bayesian algorithm. A user-based CF mechanism is adopted, and the similarity between two disparate context variables regarding their ratings as supplied by users is computed using cosine similarity for existing users. The focus in user-based collaborative filtering is to find other users similar to the active user, commonly called "neighbors". This is achieved by computing the "weight" of the active user, a, alongside every other user, concerning the similarity in their ratings given to the same items, i, and similar context(s). The results of calculating contextual information supplied by a new user are used to generate recommendations for POIs under a specific contextual situation. For existing users, ratings by similar users in similar contexts are fetched and used to calculate and recommend POIs. POIs such as hotels, restaurants, and landmarks are considered for recommendation. The system utilizes the form-filling method to collect some users' contexts, such as the user's mood, companion, and budget, for simplicity while distance_to_POI is fetched implicitly.

A major drawback of this study is using Google Maps data for the recommendation. Although Google Maps has a ton of location-based information, there are some concerns, such as incomplete data. Some POIs, particularly those smaller spots or brand-new businesses, might not appear on the map. Secondly, there is the issue of limited contextual information. Google Maps does not provide real-time details like how crowded a place is, the weather, or how popular it might be based on events around it. Thirdly, there is also the issue of dependence on Google API policies. If Google decides to change how its API works or change its pricing, data availability can become a huge problem.

To tackle these challenges, future research could benefit from looking at other data sources like Yelp, Foursquare, or even content from social media. All these could improve the coverage of POIs and add some much-needed context. Real-time updates like forecasts, social media check-ins, or event schedules could help yield more relevant and adaptable recommendations. These tweaks would make the system more dynamic, personalized, and effective for real-life situations. Future work could adopt a method of implicitly inferring these contexts from the system, possibly via Machine or Deep Learning techniques.

Author Contributions: Conceptualization: Rianat Oguntuase; Methodology, Rianat Oguntuase and Bolanle Ojokoh.; software: Rianat Oguntuase and Arome Junior Gabriel; validation: Bolanle Ojokoh and Arome Junior Gabriel; investigation: Rianat Oguntuase, Bolanle Ojokoh; resources: Bolanle Ojokoh and Arome Junior Gabriel; data curation: Arome Junior Gabriel; writing Original draft preparation: Rianat Oguntuase and Arome Junior Gabriel; writing—review and editing: Rianat Oguntuase, Bolanle Ojokoh, and Arome Junior Gabriel; visualization: Arome Junior Gabriel; supervision: Bolanle Ojokoh and Arome Junior Gabriel; project administration: Bolanle Ojokoh and Arome Junior Gabriel; funding acquisition: nil. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No new data was created.

Acknowledgements: The authors acknowledge the Federal University of Technology Akure, Nigeria, for providing the environment for the research.

Conflicts of Interest: The authors declare no conflict of interest

References

[1] F. Ricci, L. Rokach, and B. Shapira, "Recommender Systems: Introduction and Challenges," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, and B. Shapira, Eds. Boston, MA: Springer US, 2015, pp. 1–34. doi: 10.1007/978-1-4899-7637-6_1.

[2] G. Adomavicius and A. Tuzhilin, "Context-Aware Recommender Systems," in *Recommender Systems Handbook*, Boston, MA: Springer US, 2011, pp. 217–253. doi: 10.1007/978-0-387-85820-3_7.

[3] Z. Bahramian, R. Ali Abbaspour, and C. Claramunt, "A Context-Aware Tourism Recommender System Based on A Spreading Activation Method," *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. XLII-4/W4, pp. 333–339, Sep. 2017, doi: 10.5194/isprs-archives-XLII-4-W4-333-2017.

[4] K. K. San, H. H. Win, and K. E. E. Chaw, "Enhancing Hybrid Course Recommendation with Weighted Voting Ensemble Learning," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 4, pp. 337–347, Jan. 2025, doi: 10.62411/faith.3048-3719-55.

[5] A. K. Dey, “Understanding and Using Context,” *Pers. Ubiquitous Comput.*, vol. 5, no. 1, pp. 4–7, Feb. 2001, doi: 10.1007/s007790170019.
[6] S. A.El-Moemen, T. Hassan, and A. A.Sewisy, “A Context-Aware Recommender System for Personalized Places in Mobile Applications,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 3, 2016, doi: 10.14569/IJACSA.2016.070360.
[7] J. L. Herlocker and J. A. Konstan, “Content-independent task-focused recommendation,” *IEEE Internet Comput.*, vol. 5, no. 6, pp. 40–47, 2001, doi: 10.1109/4236.968830.
[8] M. Debnath, P. K. Tripathi, and R. Elmasri, “Preference-Aware Successive POI Recommendation with Spatial and Temporal Influence,” in *Social Informatics*, 2016, pp. 347–360. doi: 10.1007/978-3-319-47880-7_21.
[9] S. Sharma and D. Kaur, “Location based context aware recommender system through user defined rules,” in *International Conference on Computing, Communication & Automation*, May 2015, pp. 257–261. doi: 10.1109/CCAA.2015.7148384.
[10] M. van Setten, S. Pokraev, and J. Koolwaaij, “Context-Aware Recommendations in the Mobile Tourist Application COMPASS,” in *Adaptive Hypermedia and Adaptive Web-Based Systems: Third International Conference, AH 2004, Eindhoven, The Netherlands, August 23-26, 2004. Proceedings 3*, 2004, pp. 235–244. doi: 10.1007/978-3-540-27780-4_27.
[11] L. Baltrunas, B. Ludwig, S. Peer, and F. Ricci, “Context-Aware Places of Interest Recommendations for Mobile Users,” in *Design, User Experience, and Usability. Theory, Methods, Tools and Practice*, vol. 6769, A. Marcus, Ed. Berlin, Heidelberg: Springer, 2011, pp. 531–540. doi: 10.1007/978-3-642-21675-6_61.
[12] B. A. Ojokoh, M. O. Omisore, O. W. Samuel, and T. O. Ogunniyi, “A Fuzzy Logic-Based Personalized Recommender System,” *Int. J. Comput. Sci. Inf. Technol. Secur.*, vol. 2, no. 5, pp. 1008–1015, 2012.
[13] M. Braunhofer, M. Elahi, M. Ge, F. Ricci, and T. Schievenin, “STS: Design of Weather-Aware Mobile Recommender Systems in Tourism,” in *Proceedings of the First International Workshop on Intelligent User Interfaces: Artificial Intelligence meets Human Computer Interaction (AI*HCI 2013)*, 2013. [Online]. Available: https://ceur-ws.org/Vol-1125/paper5.pdf
[14] B. A. Ojokoh, M. O. Omisore, O. W. Samuel, and U. I. Eno, “Archetypal Personalized Recommender System for Mobile Phone Users,” *Int. J. Inf. Retr. Res.*, vol. 3, no. 3, pp. 40–58, Jul. 2013, doi: 10.4018/ijirr.2013070103.
[15] Q. Liu, “Context-aware Mobile Recommendation System Based on Context History,” *TELKOMNIKA Indones. J. Electr. Eng.*, vol. 12, no. 4, pp. 3158–3167, Apr. 2014, doi: 10.11591/telkomnika.v12i4.4786.
[16] J. Bao, Y. Zheng, and M. F. Mokbel, “Location-based and preference-aware recommendation using sparse geo-social networking data,” in *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, Nov. 2012, vol. 8, no. 3, pp. 199–208. doi: 10.1145/2424321.2424348.
[17] M. Kuanr and S. N. Mohanty, “Location-based personalised recommendation systems for the tourists in India,” *Int. J. Bus. Intell. Data Min.*, vol. 17, no. 3, p. 377, 2020, doi: 10.1504/IJBIDM.2020.109294.
[18] Q. Wang *et al.*, “Next Point-of-Interest Recommendation on Resource-Constrained Mobile Devices,” in *Proceedings of The Web Conference 2020*, Apr. 2020, pp. 906–916. doi: 10.1145/3366423.3380170.
[19] M. Kuanr, B. Kesari Rath, and S. Nandan Mohanty, “Crop Recommender System for the Farmers using Mamdani Fuzzy Inference Model,” *Int. J. Eng. Technol.*, vol. 7, no. 4.15, pp. 277–280, Oct. 2018, doi: 10.14419/ijet.v7i4.15.23006.
[20] M. Garanayak, S. N. Mohanty, A. K. Jagadev, and S. Sahoo, “Recommender system using item based collaborative filtering (CF) and K-means,” *Int. J. Knowledge-based Intell. Eng. Syst.*, vol. 23, no. 2, pp. 93–101, Jul. 2019, doi: 10.3233/KES-190402.
[21] W. Zhang and J. Wang, “Content-bootstrapped Collaborative Filtering for Medical Article Recommendations,” in *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Dec. 2018, pp. 1184–1188. doi: 10.1109/BIBM.2018.8621180.
[22] A. Chen, “Context-Aware Collaborative Filtering System: Predicting the User’s Preference in the Ubiquitous Computing Environment,” in *Location- and Context-Awareness*, 2005, pp. 244–253. doi: 10.1007/11426646_23.
[23] B. Ojokoh and A. Babalola, “A Personalized Healthy Diet Recommender System,” in *Conference: 2nd International Conference and Exhibition (OWSD-FUTA)*, 2015, pp. 388–393.
[24] B. A. Ojokoh and I. A. Amaunam, “A switching hybrid mobile recommender system for tourists,” *Int. J. Inf. Decis. Sci.*, vol. 12, no. 2, p. 176, 2020, doi: 10.1504/IJIDS.2020.106735.