Creating a Personalized Recommendation Framework in Smart Shopping by Using IoT Devices

Noura Abdaoui[1,2,a], Ismahene Hadj Khalifa[3] and Sami Faiz[2,3]
[1]ENSI, University of Manouba, Tunisia
[2]LTSIRS Laboratory, Tunisia
[3]ISAMM, University of Manouba, Tunisia

Keywords: Ubiquitous Recommender System, Personalized Recommendation, IoT, Fog Architecture, Big Data, Context.

Abstract: Personalization and recommendation are two important prerequisites that must be incorporated in the Iot environment where smart devices data are generated anywhere and anytime. Both prerequisites are essential to produce a higher satisfaction level of ubiquitous recommender system which matches the preferences of the user. Is the time to improve the quality of traditional ubiquitous recommender system which failed to exploit dynamic and heterogeneous big data in delivering personalized recommendation. In this paper, we create a framework of personalized recommendations in Smart shopping where Iot devices are connected. We proposed a Fog computing architecture to solve the ubiquitous recommendations issues related to Iot challenges. The given model is a multi-layer fog structure which aims to use the multi sources big data in order to propose personalized offers according to the users' profiles and analyze their feedbacks to improve their experiences.

1 INTRODUCTION

Ubiquitous recommender systems assist the mobile user by providing personalized recommendations of items and services in ubiquitous environment where context is the most important aspect (Mettouris and al., 2014). Nowadays, devices used in ubiquitous environment are connected to sensors coming under Iot. Most of Iot applications are connected to cloud computing. Sensors and other devices provide huge amount of data in Iot applications. They generate Big Data that will be processed and analyzed for suitable and reactive actions. Data collected by sensors must be analyzed in the cloud, which requires a high bandwidth for the network used. Thus, these issues can be solved by using fog computing (M. Chiang and T. Zhang, 2016). This new technology was invented by Cisco in 2012 as three-tier "Mobile (Iot)-Fog-Cloud" system. It widens the cloud to bring the environment close to the devices that interact with Iot data (such as user devices).

This paper is an enhanced work since the publication of (Abdaoui and al., 2018). Over there, it has proposed a basic solution to send personalized ads in ubiquitous environment after detecting the customer's localization. The motivation of this paper is how to integrate the classic ubiquitous recommendations in three-tier fog architecture. We believe that our approach can recommend personalized real time services while keeping features of fog architecture as well as enhancing traditional recommender approaches in ubiquitous computing. In brief, the paper is organized as follows. Section 2 discusses the related studies of ubiquitous computing and recommender systems related to Iot challenges, and then outlines the use of fog computing. Section 3 presents the proposed architecture in details. In section 4, we investigate the use of fog computing architecture in the ubiquitous recommender system. Section 5 describes the system's implementation and highlights the results. In Section 6, we wrap up the paper with conclusions and horizons of work that would improve the suggested ubiquitous fog-based recommender system.

[a] https://orcid.org/0000-0002-7181-0401

Abdaoui, N., Khalifa, I. and Faiz, S.
Creating a Personalized Recommendation Framework in Smart Shopping by Using IoT Devices.
DOI: 10.5220/0011969400003482
In *Proceedings of the 8th International Conference on Internet of Things, Big Data and Security (IoTBDS 2023)*, pages 200-207
ISBN: 978-989-758-643-9; ISSN: 2184-4976

2 RELATED WORK

In this section we discuss the background and related studies of ubiquitous computing in the field of recommender systems. Then, we present the main reasons behind fog computing's appearance with examples of existing recommender systems.

2.1 Ubiquitous Computing

The term ubiquitous computing, was first conceived by Mark Weiser. It is an extended sort of computational technology in the form of a microprocessor, found in every object (Mettouris and al., 2014). Recommendation in ubiquitous computing is a service which uses contextual parameters in order to provide more personalized recommendations to users. Yet, there are various challenges to the ubiquitous recommender systems. These limitations could be either technological or related to storage. While the first includes wireless technology problems and energy concerns, the latter is related to context-awareness, tacking user's intentions and privacy concerns.

2.2 Fog Computing Solution

To resolve issues mentioned previously, fog computing technology has been set forth. Fog computing is the outcome of the new requirements that have been arisen to meet the needs of the ubiquity of devices and the interest for faster management of networks and services (F. Bonomi and al. 2014). Fog computing applications processed different axes such as healthcare (Waraich, A. 2019) and smart cities (Aloqaily, M. and al., 2019). A number of fog computing challenges exists such as the simplification of mobility, reinforcement of privacy, low latency, real-time interaction, low energy consumption and the network bandwidth for real-life applications in different sectors.

2.3 Recommender System

Certainly, recommender system RS proposes items that users may prefer. Several approaches have been used in the make of RS. The collaborative filtering CF technique that recommends items based on similarity measures between users and items, as defined by (Prateek Parhi and al., 2017). (Asiri S. and al., 2016), exploits the CF method to mold an Iot trust and reputation model that investigates trust and reputation among Iot nodes. Within the same framework, Ubiquitous Context Aware Recommender Systems for Ubiquitous Learning (UbiCARsUL) is proposed by (Sukayna T. and al., 2015). The trust-aware recommender system (TARS) is therefore introduced to enhance the CF technique in Iot environment by helping users finding reliable services (Weiwei Yuan and al, 2013). But the recommender searching mechanism of TARS is still always at its beginning. It is not easy to find the most reliable recommendations for the active users in the scale-free network. In content-based filtering CBF technique, as presented by (Umair Javed and al., 2021), the algorithm recommends items and their similars that were liked in the past. SOMAR by (Zanda et al., 2012) suggests activities on Facebook from sensor data. Similarly, (Koubai, N. and al., 2019) proposes a recommender system to smart restaurants. Knowledge-based technique suggests products based on inferences about user's needs and preferences. It is based on the identified relationship between a user's needs and a possible recommendation. Another aspect like Context-aware is also used in recommendation. It uses the user's context as time and place to define the kind of recommendations. (Hassani, A., and al., 2018) set forth Context-as-a-Service (CoaaS) recommender system that uses an Iot context service to provide contextual information in smart shopping. In recent years, some attempts have been made to integrate the real-world contexts and emotions in the music recommenders like (Willian Assuncao and al., 2022). Otherwise, hybrid recommender technique as defined by (Erion Çano, Maurizio Morisio, 2017) is the one that combines multiple recommendation techniques together to produce the output with higher accuracy. Besides, (Twardowski, B. and Ryzko, D. (2015)) build an RS that uses data from both mobile devices and other Iot devices. Moreover, IBRS is an interest-based recommender system proposed by (Punam Bedi and Pooja Vashisth, 2015). It combines a hybrid RS approach with automated argumentation-based reasoning between cognitive agents. Machine learning algorithms are used also in recommender systems by many researchers namely (Sewak, M. and Singh, S. 2016), (Abdaoui N. and al., 2017) and (Ayata, D. and al., 2018).

2.4 The Cold Start Problem

The cold start problem occurs when the recommender system is unable to form any relation between users and items for which it has insufficient data due to many reasons. Low interaction between users and items as well as when a new user enrolls in the system and the recommender is required to offer recommendations for a set length of time without depending on the user's previous interactions. According to (Dina Nawara and Rasha Kashef, 2020), to provide the new user with reliable recommendations, a content-based RS should have the access to a sufficient number of user's records that allow it to determine the user's preferences. Yet, user might not receive accurate recommendations because he has very few records. Moreover, although recommending a new user's top popular offers might increase the user's purchase likelihood as it could decrease personalization. While, collaborative approach can help improve personalization, the recommendations' precision might be rather weak.

In our work, we propose a hybrid algorithm that combines the two algorithms: collaborative and content-based algorithm in order to solve the cold start problem for a new mobile user hence improve the personalized recommendation. We design the recommender model for the group of users sharing similar characteristics, then we use this model to predict the new user's recommendations.

3 UBIQUITOUS FOG BASED RECOMMENDER SYSTEM ARCHITECTURE

While cloud computing has been widely used in shopping center, it cannot cope with the challenges arising in many Internet of Things scenarios, such as network bandwidth constraints and constrained devices. Fog architecture is a promising and effective solution to these challenges to serving mobile users and Iot devices by proactively catching and processing the required data. Also, we propose hybrid algorithm that combines the two algorithms: collaborative and content-based algorithm in order to solve the cold start problem for new mobile user and improve the personalized recommendation. In the proposed architecture, each mobile user is connected with different fog nodes in different floors through wireless access technologies WIFI. The fog server can be interconnected by wireless communication technologies. Each fog server is linked to the cloud by IP core network. This architecture provides efficient data processing and storing services. Each fog node represents Ubiquitous fog-based RS that has a number of mobile users' interfaces connected to both layers: things layer and the cloud layer. The system uses virtualized machines with multiple VMs running under a highly capable hypervisor. That hypervisor includes real-time enhancements. The Ubiquitous fog-based RS provides personalized and localized recommendations to mobile user in real time. There is a basic Linux host operating system, as there are different modules of recommendation and extensions for real-time operations and enhanced security. Many resident modules are parts of the Ubiquitous fog-based RS, including data management, recommendations, results display and identification with active mobile user. Therefore, hybrid algorithms are implemented. The goal is to learn contents from user's profile. Then, according to the required information and his contextual information interacting with fog nodes, we provide personalized recommendations in real time. More details about the Ubiquitous fog-based RS are discussed in the next section.

4 UBIQUITOUS FOG-BASED RS MODEL

As shows Figure 1, there are four principal components in our proposed system: mobile user profile, Ubiquitous fog user interface, Process of ubiquitous fog recommendations and fog data processing.

4.1 Mobile User Profile

The user's profile can be extracted from many sources namely:

Inscription: through a registration form; preferences; ratings on the items consulted and demographic attributes. Demographic data can be used to calculate recommendations for new mobile users. It used to solve the cold start problem.

Consultation: "even without log-in": analysis of mobile user's behavior implicitly. We call these behaviors "traces of use" such as "copying/pasting", searching for a product on a page and navigation indicators, such as frequency and duration of browsing, number of clicks and mouse hovers on a page or links, scrolling, etc.

Context: is the integration of contextual information (location, time, physical environment, …) for the generation of dynamic and personalized visit itineraries.

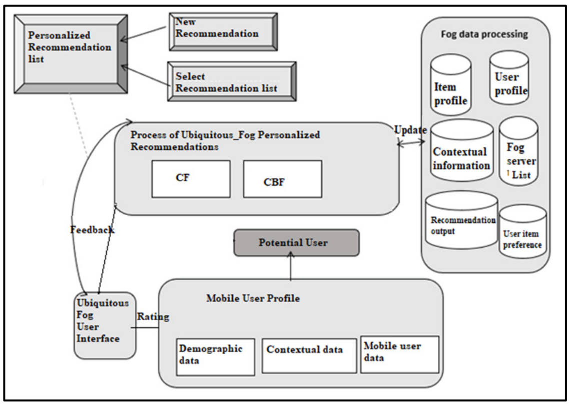

Figure 1: Ubiquitous fog-based RS Model.

4.2 Ubiquitous Fog User Interface

It supports interactions between the potential user and the system of recommendations that include many functions. The Ubiquitous fog user's interface displays the recommended item to active user and the reasons for the recommendation to explain why that item has been selected to the potential user. It will accompany the mobile user during all phases of his/ her visit to a shopping center: "before/during/after". He starts by researching brands and items to collect information and make choices. The system matches the itinerary with services and personalized content.

4.3 Process of Ubiquitous and Fog Personalized Recommendations

In this module, we aim at providing personalized recommendations to the users. The inputs of the process are user, item, user-item preference, contextual information, fog server id, workload capability, storage capability and processing capability. The output in most cases is directly shown to the potential user by the Ubiquitous fog user's interface. The hybrid algorithms used in this module combines Collaborative Filtering with Content based filtering algorithms.

4.3.1 Collaborative Filtering

A collaborative filtering system collects and analyzes a user's behavior based on a user's preferences given in the form of feedback, ratings, and other interactions. More specifically, a user-item rating matrix of preferences for items by users is constructed. From this matrix, user's matches are made on the basis of finding similar preferences and interests by calculating similarities between user's profiles. The Equation (1) expresses how to calculate the similarity between two users, where Vaj expresses an assessment made by the active user a on a product j, V_{ij} the one made by the user u_i , vvi the average of the ratings of the user u_i and vv_a the average of the valuations of the active user.

$$Sim(u_a, u_i) = \frac{\sum_j (v_{aj} - vv_a)(v_{ij} - vv_i)}{\sqrt{\sum_j (v_{ij} - vv_i)^2 (v_{ij} - vv_i)^2}} \qquad (1)$$

After matching between any products and users, we try to find the k user (neighbors) with the highest coefficient of resemblance to the active user. This prediction is obtained from the weighted sum of the valuations of other users using the following Equation (2)

$$P(u_a, u_i) = rr_a \frac{\sum_{u\in K}(r_{u,i} - rr_u) \times Sim(u_a - u_j)}{\sum_{u\in K} Sim(u_a - u_j)} \qquad (2)$$

Then, there is the prediction of the valuation that the active user $P(u_a, u_i)u_a$ would assign to the item i. The average rating of the user is rr and is the similarity between the active user and the user $Sim(u_a - u_j)i$. K is the subset of similar k-users while $r_{u,i}$ is the rating of the neighbor u to item i .

4.3.2 Content-Based Filtering

In Ubiquitous fog-based RS, similar items are proposed as part of the recommendations. This technique has an advantage of providing recommendations to the user with an item which has not been rated yet. It provides a potential user's independence: by focusing only upon the dynamic user's ratings. It also provides transparency by posting the characteristics of an item explicitly from the list of recommendations.

4.3.3 Hybrid Filtering

Content-based filtering technique does not involve the opinions of all users when recommending items are consequently limited to making recommendations that are in the range of a user's likes. However, collaborative filtering cannot provide predictions to items that have not yet been rated, commonly known as the cold start problem. Therefore, hybrid filtering techniques overcome these limitations and use a combination of content-based and collaborative techniques to improve performance. The idea is that the resultant algorithm will provide more accurate and effective recommendations than any single algorithm. Given a new user, we do not intend to find a single similar user, but we look for a group of users with similar characteristics instead. Then, we do not directly recommend to new users the offers that have been bought by similar groups. Would rather, we use a content-based method to build the recommender's model for the group and use the model to predict the new user's recommendations.

4.3.4 Personalized Recommendations

The filtering rules are applied. The result of a personalized recommendation list is displayed to Ubiquitous fog user's interface taking into consideration not only the user's advanced and basic profile, but also her/his customized preferences and feedbacks. Assume that there is a set of instances X : {X1,X2… Xn}and a binary preference function pref(i,j,fh)>0 means i is preferred to j by user u in Floor Fh pref(i,j,fh)= 0 means that neither of the two items is preferred, and pref(i,j,fh)< 0 means j is preferred to j by user u in Fh. shown in Equation 3. Nu a set of users with similar preferences to those of target user,$\boldsymbol{N_u^{I,J,f}}$ is the set of neighbors of u who rated items i and j in the same fog server floor F.

$$\boldsymbol{pref(i,j,F_h)} = \frac{\sum_{v\in N_u^{I,J,f}} \boldsymbol{sim_{u,v,F}}\left(\boldsymbol{r_{v,i,F}} - \boldsymbol{r_{v,J,F}}\right)}{\sum_{v\in N_u^{I,J,f}} \boldsymbol{sim_{u,v,F}}} \tag{3}$$

$sim_{u,v,F}$ is the similarity between u and v at the same level of fog server F ; and denotes user's ratings of items i and j at the same level of fog server F.$sim_{u,v,F} r_{v,i,F} r_{v,j,F}$. The process can be summarized as following:

1. If a user exists, acquire input from the content-based model
2. Produce a relevance score for all users and all products.
3. Execute the collaborative filtering to generate predicted ratings for all users.
4. Filter predictions.
5. Affect weighting factors for the content-based model and collaborative filtering model to maximize performance.
6. Turn back predictions in a descending order, relevant items are at the top.
7. If user does not exist, the system insists to enter user's details such as age, sexe,etc.
8. A user's information goes through a popularity-based algorithm. This model finds the most popular products by considering the average rating for products and the maximum number of ratings per product. A list of similar products is then sorted in a descending order, filtered out and represented as recommendations to the user.
9. Providing feedback on the quality of a recommendation list in the form of ratings. This feedback is saved for readjusting the weights and recalibrating the recommender model.

To solve the cold start problem, we design the user's personal preference model that takes as an input the historical data of a single user and as in output the user's contextual preference. For the newly registered users, the system has no historical data. Therefore, a clustering block is designed using K-means and Density-based spatial clustering of applications with noise (DBSCAN) (A. Smiti, Z. and Elouedi, 2012) algorithms. These algorithms are used easily with any data type. We employ the Elbow method and the Silhouette coefficient in (James, G. and al., 2013) to get the optimal number of clusters. For each cluster, it collects the historical data of all users belonging to that group. Next, the classification module is used to learn about a group-level contextual preference model. Having a clustering model and group-level contextual preference models for groups in hand, as soon as new user registers to the system. Finally, the group-level contextual preference model will be used as the user's personal preference model. The model will be in use until the system collects enough historical data from the new user to build a pure personal contextual preference model.

4.4 Fog Data Processing

In this module, raw datasets are filtered, converted and stored into various databases. The aim of data filtering is to extract useful information to the recommender system. We categorize six databases that describe a way of identifying the dataset. As shows figure 1:

- Item Profile: this database contains item attributes, such as item ID, description and category, and virtual content size.
- User's profile: this database contains user's attributes, such as age and gender.
- User-Item Preference: this database contains user-item preference information that has been converted from raw data. A preference could either be explicit or implicit.
- Fog Server List: fog server ID, workload capability, processing capability, storage capability and power usage.
- Contextual Information: this database contains contextual information such as user's

localization, time, and network bandwidth information.
- Recommendation Output: this database contains the final output. It stores the output generated during the process of creating a recommendation, such as similarity matrix, training data and testing data.

The basic idea is to maximize the use of a fog server closer to the user. We try to process and store data as close as possible to the user who is connected to the target fog server. If the target fog server cannot provide such a service, it will send a request to neighboring fog servers. If neighboring fog servers cannot provide the requested service, then the request will be sent to the cloud.

5 EXPERIMENTS

In the following section, we implement the Ubiquitous fog recommender system in smart shopping by deploying five fog servers. We performance a set of experiment based on a real-world dataset.

5.1 Integration of Ubiquitous Fog-based RS in Smart Shopping

Ubiquitous fog recommender system is evaluated using a dataset containing ID address, destination IP address, connected fog server IP address, and the localization of the mobile user. We treat each ID address as a user, because it is a unique identifier of his Mobile in our fog environment. An item could refer to Idsensor connected to a product or a web site visited by a person. In our use case, we have used many types of nodes in the three floors. Static nodes such as beacons attached to different products and fog nodes in the different floors. Also, we use mobile nodes such as mobile phone. Each mobile user is identified by his mobile's API. Mobile user sensor is detected by mobile devices which contact the fog node with the proximity information. We have deployed also the fog-based hybrid recommender system on each fog server and an Alibaba cloud server. The mobile user can access to the Internet and use smart devices by connecting to the corresponding fog server. The dataset is obtained from the deployed fog servers.
We have used different platforms to simulate fog servers. A typical platform is Windows10 OS with Intel Core i5 CPU@2.7 GHz and 16 GB memory. All algorithms were implemented using python with the following installed libraries: NumPy, pandas, matplotlib, scikit-learn, nltk, scipy. Anaconda has been used for python package management and deployment. We collect 200 records of different users' interaction with different items in the mall. All data are obtained from the deployed five simulated fog servers. Due to the small amount of data set, we use all the data and split data into 80% for training purpose, 20% as test data set.

5.2 Evaluation Data Utility vs RMSE and MAE

In the following experiments, each floor which includes fog server is the target level. Any user connected to fog servers deployed on these floors is considered in location. In the experiments, we vary the weight value w_j and attempt identifying the best value of each parameter to obtain the most accurate result. We use two popular evaluation methods for recommendation, mean absolute error (MAE) and root-mean-square error (RMSE), to justify our quality of the prediction. RMSE (Equation 4) penalizes large errors by amplifying the differences between the predicted preferences items and the real ones:

$$\boldsymbol{RMSE} = \sqrt{\frac{\sum(\boldsymbol{test} - \boldsymbol{rsl})^2}{|\boldsymbol{test}|}} \tag{4}$$

MAE (Equation 5) is the average absolute deviation of the predicted ratings from the real ratings of items:

$$\boldsymbol{MAE} = \frac{\sum|\boldsymbol{test} - \boldsymbol{rsl}|}{\boldsymbol{test}} \tag{5}$$

We set up a decay factor for the weight parameter wj to observe its impact on the prediction accuracy and data utility. Here, the decay factor has been set up as $(\frac{wj}{n})$ for present location, the weight of the third level of location is$(\frac{wj}{n})^3$. wj is the weight at level j that controls prediction at each location level and affects the final prediction. The weights satisfy the following constraints: wj ϵ [0,1], w1+w2…+wj=1. In figure 2, wj is fixed as 0.7, n varies from 1 to 4. This value has also been mapped to the amount of privacy levels, thus there are 4 privacy levels from ϵ [0, β] based on location. In Figure 2, we observe that all three measurements show a similar trend. The value of data utility decreases from 1.3 to 0. 5. The RMSE and MAE value both decreases. RMSE decreases from 3.5 to 0. 77. MAE decreases from 1.7 to 0. 5. If wj is higher, both MAE and RMSE results are better. If n is higher, both evaluation results are better as well. However, the trend becomes weaker.

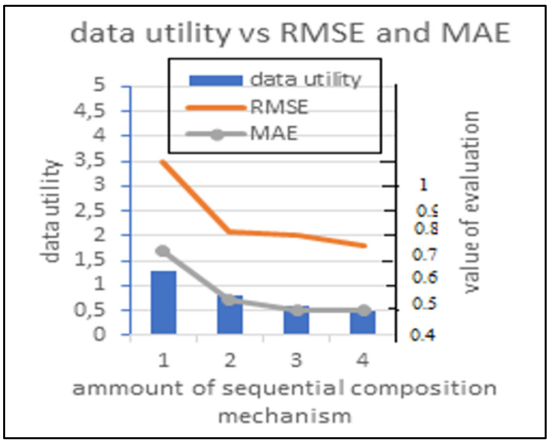

Figure 2: Data utility vs RMSE and MAE.

5.3 Evaluation Values of Precision and Recall

Here, also each floor which includes fog server is the target level. w_k is the weight value of the target level. As a result, we need to be aware of the impact of recommended content size on the server. We use the recall and precision method to measure our result. Both methods (as defined in (6) and (7)) are broadly used in evaluating information retrieval and statistical classification. In general, precision represents the prediction accuracy, while recall represents the prediction scale. Ideally, both values would be high.

$$P = \frac{Number\ of\ correct\ items\ recommended}{Total\ number\ of\ items\ recommended} \quad (6)$$

$$R = \frac{Number\ of\ correct\ items\ recommended}{Number\ of\ relevant\ items\ recommended} \quad (7)$$

We vary the weight value w_k from 0 to 1 with the step size of 0,2 to observe the impact of result accuracy. We also modify the number of prediction items from 4 to 10 to observe the impact of result accuracy. In Figure 3, each band represents the change of the w_k from 0 to 1. The height of each point on a given band is the evaluation value of recall. The length between two points on a given band represents the difference between two results. Various bands represent different numbers of predictions, corresponding to items recommended to a fog server. A higher value of the prediction number means requiring more storage space on a fog server. Figure 3 shows that the more items are predicted, the higher the recall value is.

In Figure 4, the height of each point on a band represents the evaluation value of precision. We also observe that the more items are predicted, the lower the precision value. If w_k is 0.3, we obtain most accurate result on each band. So, if w_k is 0,3, we obtain the best evaluation results for both precision and recall. The prediction number does not impact the trend pattern of the evaluation value.

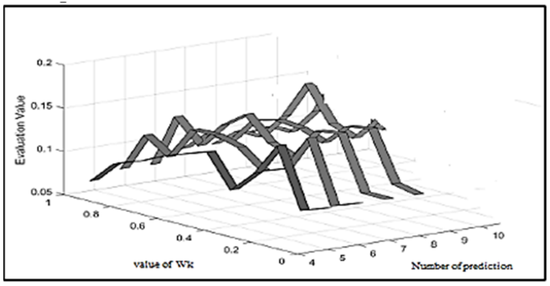

Figure 3: Evaluation value of recall.

However, the more items are predicted, the worse the predicted results are, and the higher the recall is.

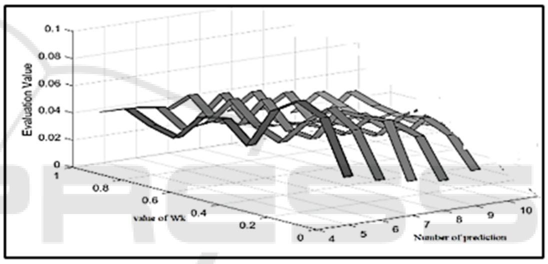

Figure 4: Evaluation value of precision.

The obtained results prove that the efficiency of Ubiquitous fog-based RS and its sample algorithms are feasible and can run independently from the cloud server. The system helps fog servers choose the most frequently requested content to purchase in order to save bandwidth, storage resources and used network resources. It provides also much more accurate recommendation results for certain items based on fog server location.

6 CONCLUSIONS AND FUTURE LINES

The system is designed to address the problem of information overloading ubiquitous environment, and thus to become a tool of fog computing optimization. We have reviewed the state of the art of ubiquitous recommender system in several sectors then the reason behind fog computing integration. The proposed system of recommendation improves the user's experience in the smart shopping center where we have used Iot devices. Experimental results demonstrate that Ubiquitous fog-based RS provides highly accurate and personalized recommendations to mobile users. It considers the fog server as well as contextual data of mobile user. Furthermore, it incorporates feedbacks collected from mobile users. Adding to that, it improves customers' experiences stored in the server and anticipates new users' needs. In future research, we intend to extend our proposal to areas with deep learning algorithms and reinforcement learning which can be used to improve the current research and overcome limitations.

REFERENCES

Abdaoui Noura, Ismahène Hadj Khalifa, Sami Faïz (2017). Process of Personalizing the Ubiquitous Advertisements. *International Journal of Knowledge Society Research (IJKSR)*, 8 (2), pp.13-35.

Abdaoui Noura, Ismahène Hadj Khalifa, Sami Faïz (2018). Improved Indoor Ge-olocalization System Between Bluetooth Low Energy Technology of iBeacons and Wi-Fi Network for Better Iot Systems Accuracy, in Knowledge_Intensive Economies and opportunities for Social,Organizational, and Technological Growth. *International Journal of Smart Education and Urban Society (IJSEUS*), IGI-global, chapter 9.

A. Smiti, Z. Elouedi (2012), Dbscan-gm: An improved clustering method based on gaussian means and dbscan techniques. *2012 IEEE 16th International Conference on Intelligent Engineering Systems (INES)*, pp. 573–578.

Aloqaily, M. , Al Ridhawi, I. , Salameh, H. B. , & Jararweh, h. (2019). Data and service management in densely crowded environments: Challenges, opportunities, and recent developments. *IEEE Communications Magazine*, 57 (4), 81–87.

Asiri, S. and Miri, A. (2016). An iot trust and reputation model based on recommender systems. *In Privacy, Security and Trust (PST), 14th Annual Conference*, pp. 561–568. IEEE.

Ayata, D., Yaslan, Y. and Kamasak, M.E. (2018). Emotion based music recommendation system using wearable physiological sensors. *IEEE Transactions on Consumer Electronics*, 64, 196– 203.

Dina Nawara, Rasha Kashef (2020). IoT based Recommendation Systems _An Overview. *IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS).* pp 1-7.

Erion Çano, Maurizio Morisio (2017). Hybrid recommender systems: A systematic literature review. *Intelligent Data Analysis.* Volume 21. Issue 6.pp 1487–1524.

F. Bonomi, R. Milito, P. Natarajan, and J. Zhu (2014). Fog computing: A platform for internet of things and analytics. *Big Data and Internet of Things: A Roadmap for Smart Environments*. pp. 169–186.

Hassani, A., Haghighi, P.D., Ling, S., et al. (2018). Querying iot services: A smart carpark recommender use case. In *Internet of Things (WF-IoT), 2018 IEEE 4th World Forum on*, pp. 619– 624.

James, G.; Witten, D.; Hastie, T.; Tibshirani, R (2013). An Introduction to Statistical Learning; *Springer: Berlin, Germany; V*olume 112.

J Li, Z He, Y Cui, C Wang, C Chen, C Yu, Min Zhang, Yiqun Liu, Shaoping Ma. (2022). Towards Ubiquitous Personalized Music Recommendation with Smart Bracelets. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies.* pp.1–34

Koubai, N. and Bouyakoub, F. (2019). Myrestaurant: A smart restaurant with a recommendation system. *International Journal of Computing and Digital Systems,* 8, 143–156.

M. Chiang and T. Zhang (2016). Fog and iot: An overview of research opportunities. *IEEE Internet of Things Journal,* vol. 3, no. 6, pp. 854–864.

Mettouris Christos, and George A. Papadopoulos (2014). "Ubiquitous recommender systems." Computing 96.3. 223-257.

Prateek Parhi, Ashish Pal, Manuj Aggarwal (2017). A survey of methods of collaborative filtering techniques. 2017 International Conference on Inventive Systems and Control (ICISC).

Punam Bedi & Pooja Vashisth (2015). Argumentation-enabled interestbased personalised recommender system. *Journal of Experimental & Theoretical Artificial Intelligence*,Volume 27, pp199-226. Issue 2

Sewak, M. and Singh, S. (2016). Iot and distributed machine learning powered optimal state recommender solution. *In Internet of Things and Applications (IOTA), International Conference on*, pp. 101–106. *IEEE*.

Twardowski, B. and Ryzko, D. (2015). Iot and context-aware mobile recommendations using multi-agent systems. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT),* pp. 33–40. *IEEE.*

Umair Javed, K. Shaukat, I. Hameed, Farhat Iqbal, Talha Mahboob Alam, S. Luo (2021). A Review of Content-Based and Context Based Recommendation Systems. *Int. J. Emerg. Technol. Learn.*

Waraich, A. (2019). Remote health monitoring of elderly through wearable sensors. *Multimedia Tools and Applications*, 1–26.

Weiwei Yuan, Donghai Guan, Lei Shu & Jianwei Niu (2013). Recommender Searching Mechanism for Trust Aware Recommender Systems in Internet of Things, in *Automatika*, Volume 54, - Issue 4.

Willian Assuncao, Lara S. G. Piccolo, and Luciana A. M. Zaina. (2022). Considering emotions and contextual factors in music recommendation: a systematic literature review. *Multim. Tools Appl.* 81, 6, 8367--8407.

Zanda, Andrea, Santiago Eibe, and Ernestina Menasalvas (2012). SOMAR: A social mobile activity recommender. *Expert Systems with Applications* 39.9. 8423-8429.