 DOI 10.1109/ICCCN52240.2021.9522241

An Edge-based Strategy for Smart Advertising

Rasool Seyghaly
UPC BarcelonaTECH
CRAAX Lab.
Vilanova, Spain

Jordi Garcia
UPC BarcelonaTECH
CRAAX Lab.
Vilanova, Spain

Xavi Masip-Bruin
UPC BarcelonaTECH
CRAAX Lab.
Vilanova, Spain

***Abstract*—Smart advertising creates awareness about some offer with a more direct, personalized and interactive focus. In this area, AROUND is a social network aimed at providing smart advertising to suggest appealing business to their customers and friends. The AROUND system is supported by a sophisticated recommender system, which considers not only the customers historical behaviours, but also their current mood and accurate location. In such smart recommendation systems, the response time for the personalized advertising is critical for a successful users' quality of experience. In this research work we first evaluate the current performance of the AROUND system in terms of processing and communication times considering that, nowadays, this social network has more than 3 million users. The current implementation of the system relies on the deployment of a network of beacons, and uses a domestic cloud provider as the main infrastructure. We show that when the number of concurrent requests becomes too high, the response time faces some limitations. In order to address this issue, we discuss several alternatives, and propose the use of an edge-based strategy as a solution for fast response time. In the experimental section, we measure the performance of the AROUND system, both in our current infrastructure at the cloud and with an edge-based approach, and show the additional advantages of leveraging the edge-based strategy even in the case of overloading the cloud capacity.**

I. Introduction

It is reported by the United Nations that 55% of the world population resided in urban areas in the year 2014, and by 2050 this figure is expected to increase to 68% [1]. The migration of human beings from rural to urban areas, mainly in developing economies, as well as the increase in the World's population, create huge pressure on the existing framework of current cities. Cities all over the world have reached their existential limit. Continuous urbanisation poses severe challenges on sustainable development and living quality of urban residents. The vision of smart cities is to make more efficient uses of scarce resources, and to improve quality of citizen lives and public services [2].

Integration of all smart environments with the IoT devices (such as sensors, actuators, and smartphones) in the city can play a vital role to develop the urban services by building their city digital and smarter. Thus, it is a challenging task to integrate IoT devices and smart systems in order to harvest and process such big amount of real-time city data in an effective manner [3]. With the development of the IoT, embedded devices can be built into every fabric of urban environments and connected with each other. Data generated by these devices can be preprocessed, integrated, and made available in standard formats through open services, and applied to a variety of city areas, such as smart mobility, smart transport, smart grid, smart energy, or smart people, just to name a few.

One area in which smart services are growing in popularity but, mainly, in business impact, is smart advertising. Different than traditional advertising, smart advertising creates awareness about some product with a more friendly, interactive, personalized and creative focus. Smart advertising can be defined as advertising with intelligence aiming to create incredible experiences for the customers. Digital advertising is one of the fastest growing industries all over the world. Today, mobile advertising accounts for a large share of open source digital advertising, reaching a 51% share of the whole digital market. On the other side, the advertising ecosystem faces a major threat from ad frauds caused by false display requests or clicks, generated by malicious codes, bot-nets, click-firms, etc. Around 30% revenue is being wasted due to frauds [4].

As part of the broad spectrum of smart advertising systems, the AROUND application is a social network with focus on small and midsize business [5]. With more than 3 millions users, the main approach of this application is smart advertising with the support of IoT sensors. Specifically, multiple bluetooth low energy beacon devices are deployed in different locations in commercial shopping malls in order to track their customers and offer personalized interactive functionalities to advertisers. Users with the app can define their business and personal preferences and, when entering the commercial mall, they can choose their mood between five different states. The AROUND system has a smart recommender system which individually suggests each user appealing business and friends, considering their profile, history, current mood, time spent at a location like shops, or their current location.

The current implementation of this application is based on the cloud. All positioning data from the beacon is detected by the users' app installed in their smartphone, which sends them to the cloud. These data are combined with the users' profile, as well as the current advertising campaigns, and generates the personalized feedback for the users. However, as the number of users in each beacon range covered area grows, the volume of data and personalized requests increase respectively. According to this, the communications round-trip and processing time in cloud for suggestions may take longer than desirable, and reduce users satisfaction. The response time of the AROUND smart system has to be minimized in order to offer the end users a low latency experience.

The emergence of the Internet of Things (IoT) paradigm has led to the rise of a variety of applications with different characteristics and Quality of Service (QoS) requirements. Those applications require computational power and have time sensitive requirements. Cloud computing provides an illusion to consumers with unlimited computation resource power; however, cloud computing fails to deliver the time-sensitive requirements of some applications. The main challenges in the cloud computing paradigm are the associated delays from the IoT device at the edge to the cloud data center, and the way back. Fog computing [6], or edge computing [7], are technologies to extend the computational capabilities closer to the edge device to guarantee the time sensitive requirement of smart applications.

In this research work we present the AROUND smart advertising service and discuss a new architectural strategy for achieving the desired response time. The architecture proposed is based on the edge computing paradigm, aimed at minimizing the service's latency as well as allowing localizing the system business model, enhancing even more the overall system performance and quality of experience. In the experimental section we compare the performance of the edge-based approach with respect to the cloud, even in the case of overloading the cloud capacity, and show the additional advantages of leveraging the edge technology.

The rest of the paper is organized as follows. Section 2 reviews the related works in the areas of smart advertising and IoT-based recommendation systems. In Section 3 the AROUND system is described, and in Section 4 the current system limitations are presented. In Section 5 different solutions to overcome such limitations are discussed. And Section 6 shows the experimental analysis. Finally, Section 7 concludes the paper highlighting the main research findings.

II. Related Work

The emergence of the IoT paradigm has boosted the raise of all sort of smart services, including smart advertising. In this area, several researchers have been contributing on this topic and proposing different solutions. For instance, Kim and Lee [8] suggested that the mobile advertising paradigm is shifting and pursuing a customized and context-aware advertisement service for each consumer. The purpose of their study was to discover customer typologies through a combination of the Q and R empirical methods, and use them as a basic statistical data for advertising marketing and customer relationship management domains. Verhoef et al. [9] noticed that today's consumers are immersed in a vast and complex array of networks, featuring an interconnected mesh of people. They introduced the POP-framework, discussing how People, Objects, and the Physical world inter-connect with each other, and proposed examining the potential impact of IoT and smart products on the consumer behavior and firm strategies. Helberger et al. [10] realized that the advertising message creation, targeting, and delivery is taking whole new forms, processes, and routes, and that smart advertising is transforming the roles of consumers. Today's consumers are taking on the roles of advertisers, creators of ad content by actively engaging with brands, and active distributors of ads spreading messages created by advertisers, as well as consumers themselves, through interpersonal networks. This means that consumers are far more influential than ever before. Levanov et al. [11] proposed several algorithms for smart advertising and presented a prototype with some basic functionalities, such as a client-server application to obtain data of the user's interests and determine the user's location through the smartphone to display the most relevant advertising for that particular user. All these works have paved the way to modern smart advertising, and addressed some sort of smart advertising prototypes; however, none of these makes use of sophisticated and comprehensive recommendation systems nor use edge technology to provide more efficient localized interaction.

In this sense, one technological core element in our technology is the users' recommendation system based on their behaviour and location. Recommender systems are information filtering processes designed to ease decision-making in domains and applications where there are many options to choose from. Several authors have also contributed on recommenders based on the users' behaviour and location. For instance, Siryani et al. [12] presented a data-driven DSS to improve electric smart meter operations within the IoT ecosystem. This methodology enables prediction decisions about whether to send a technician to a customer's site or resolve the case remotely, using and comparing four different ML techniques. They demonstrated the efficiency of their approach with a complete Bayesian network prediction model. Ryu and Park [13] proposed applying the persuasion knowledge model (PKM) for consumers, based on their persuasion knowledge and their understanding of how location-based advertising works, which affect their assessment of benefits and harms. Furthermore, they examined different dimensions of persuasion knowledge by exploring to what extent the objective and subjective persuasion knowledge have differential impacts on consumers' benefit. Abbasi-Moud et al [14] introduced a tourism recommendation system that extracts users' preferences in order to provide personalized recommendations. To this end, users' reviews on tourism in social networks are used as a rich source of information to extract preferences. Then, the comments are preprocessed, semantically clustered, and sentimentally analyzed to detect a tourist's preferences. The system utilizes the vital contextual information of time, location, and weather to filter unsuitable items and increase the quality of suggestions regarding the current situation. The proposed recommendation system is developed on a dataset gathered from TripAdvisor platform. Finally, a comprehensive survey about the existing literature on recommender systems as well as a taxonomy for these systems has been presented by Quijano et al. [15]. Similar to other recommendation systems, the AROUND system suggests recommendations through a trained model. However, our model is considering a rich set of users features, such as their preferences, their contacts in the social network, their shared posts but, most importantly, their daily mood, as well as an accurate positioning with respect to the network of registered beacons deployed around the city. We believe that using beacon technology is an effective approach to detect users and accurately monitor their location and behavior.

Finally, our main challenge in AROUND is the system's response time. Achieving low delay in receiving the targeted advertisement is critical for keeping a low latency users' experience and providing a successful quality of service. Several related works are also concerned on achieving low latency. For instance, Bai and Cambazoglu [16] proposed a large-scale analysis using query logs obtained from a commercial web search. Their research is based on big data and they have analyzed the short-term and long-term impact of search response latency. This analysis demonstrates the importance of serving sponsored search results with low latency and provided insight into the digital environment and electronic advertising to ensure long-term user engagement. Nayebi and Abran [17] shew the rapid increase in the number of applications and total app store revenue which accelerated data mining and opinion aggregation studies. While development companies have pursued upfront opinion for business intelligence and marketing purposes, research interest into app ecosystem and user reviews is relatively new. There are now some academic studies addressing various challenges, such as reducing response delays and usage. The authors of this study have thoroughly examined these challenges and listed the results in categorized cases. Ameen et al. [18] provided an overview of literature addressing consumer interaction with cutting-edge technologies and explored and different challenges. They proposed six main areas for future research namely: rethinking consumer behaviour models, identifying behavioural differences among different generations of consumers, understanding how consumers interact with automated services, speeding up requests and short delays, ethics, privacy, and the consumer security concerns with new-age technologies during and after a major global crisis such as the COVID-19 pandemic.

In all the cited researches, various methods were considered for smart advertising that work on the basis of smart environments and IoT, positioning, data analysis, neural network, playing a significant role in satisfying the users experience and improving the smart services. Different from previous research works, the AROUND system provides a more advanced recommender technology, leveraging the information obtained from the customers' social network, as well as the defined daily mood. Our system also takes advantage of an accurate positioning technology through the use of beacons deployed in all registered business. With all these features, a complex model is trained to classify users into a comprehensive set of categories and, therefore, providing accurate recommendations. Finally, we also propose the use of edge computing technology as the best option for providing an effective, yet efficient, advertising system. All these details are further detailed in the next section.

III. The AROUND Application

The AROUND application is a social network with focus on small and midsize business through a smart advertising system. The aim of AROUND is connecting customers with traders by framing personalized ads in order to offer optimized shopping experience and improve the potential business between both sides [5]. Currently, the AROUND system has more than 3 million users (one million active users) and 12,000 beacons deployed in Iran and Russia.

For this reason, the AROUND system defines two users types: traders and customers. On the one hand, traders are shop owners who establish a collaboration with the AROUND system and propose a set of personalized offers and advertising options according to some sort of users' profile. They also install some beacons in different places to allow localized interaction with their customers. For instance, Fig. 1 shows the distribution of beacons in the city of Tehran, with more than 800 registered traders and more than 3,000 beacons deployed around the city.

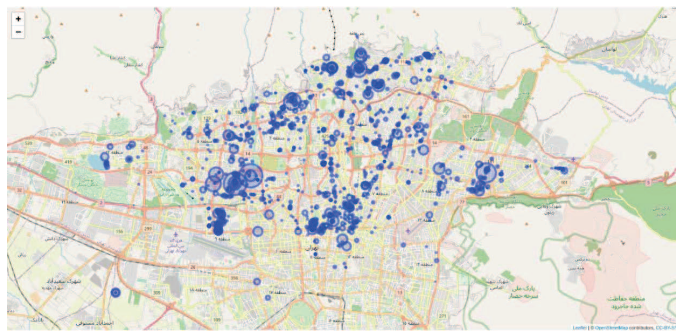
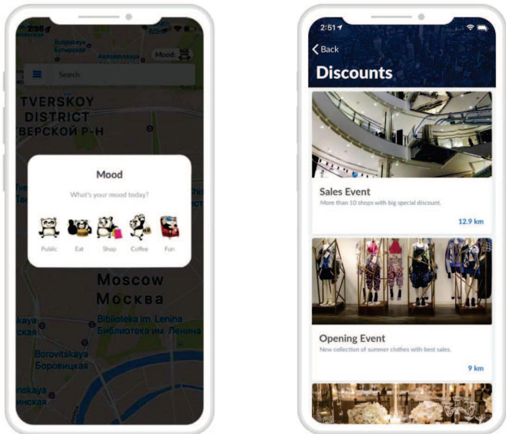


Fig. 1. Map of current beacon deployment in the city of Tehran.

On the other hand, customers are app users who wish to enjoy in the social network, exchange experiences with other users, and benefit from a personalized shopping experience. On registration, customers define their profile and, every day, they can choose between five different moods, as shown in Fig. 2.a. The main service for the AROUND users is to suggest the best advertising considering their historic behaviour, which is determined by analyzing their in-app behaviour and current location, and combined with their current defined mood. As an example, Fig. 2.b shows an advertisement sample.

The current AROUND system operation is illustrated in Fig. 3. Several beacons are deployed through the registered businesses, such as shops, museums, or any other commercial activity. When a registered customer walks close to that business, the app in the smartphone detects the beacon (1) and connects to the AROUND server (2), which is in the cloud, indicating the current customer's location (derived from the device location and the beacon ranged status). With this information, together with the client profile, the AROUND system executes the trained model and decides the most suitable advertisement for that customer (4). The personalized promotions and discounts are sent back to the customer (6) and viewed through the smartphone app.



(a) (b)

Fig. 2. Screenshot of the mood selection (a) and sample advertisement (b) in AROUND.

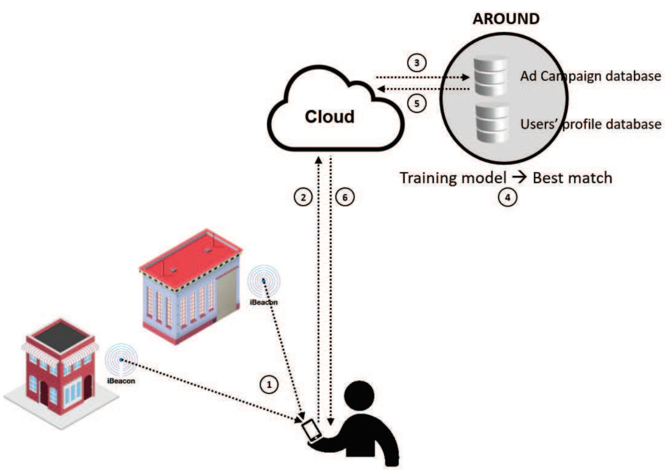


Fig. 3. Basic operation of the AROUND system.

The decision process is as follows. For each user ranged by a beacon, a request is sent to a docker instance in a Kubernetes orchestration system [19]. There is a clustering model to classify the request in 9 main categories, with additional predefined scenarios for each category. The system considers the users' different behaviours, their age, their past activity, and the interaction history with other users, as well as the different socioeconomic districts. After conditioning on their current mood, the trained model suggests which category fits them best, and decides the personalized advertisement for that user. The AROUND system databases and business model is currently running centrally in the cloud.

IV. Problem Description

The core business of the AROUND system is the smart recommender system for our end users, which suggests each user appealing business and friendship interactions based on their history and their current mood. This technology is appropriate as long as the time for recommendation fits within some predefined bounds. For instance, response times higher than 9 milliseconds can be sensed by humans; with a little bit of compensation users can accept 3 seconds for a smart recommendation system. For this reason, the response time of our smart system has to be minimized in order to offer the end users a low latency experience.

Under low requests load, the current implementation of the AROUND recommendation system works within the expected performance. However, as the number of customers increase, the response time of the pre-trained model increases as well and, therefore, the user experiences a long delay for the targeted advertisement. In order to illustrate this phenomenon, we have measured in the current AROUND system the round-trip response time since the user detects the beacon till the personalized advertisement is received. In this illustrating experiment, we have created a script to launch a number of concurrent requests ranging from 100 to 4,000. Fig. 4 shows the total response time, broken down into communication time and processing time (in seconds).

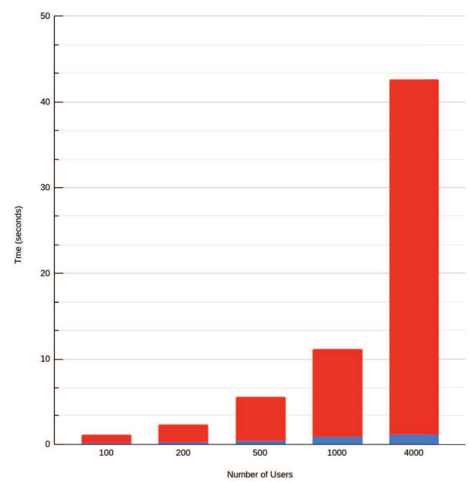


Fig. 4. Total response time, including communication and processing times.

As can be observed from the figure, as the number of users in a certain area grows, the volume of concurrent requests increases respectively and suggestions take much longer. In addition, the main weight for the response time is in the processing component, although the communication time between the customers and the cloud becomes significant for 4,000 concurrent users, reaching more than one second. Table I shows the detailed values for these measures. Note that when the system has 500 concurrent users, or more, the response time is above 5 seconds, what turns the system response time into impractical.

Currently, there are approximately 1 million active users in the AROUND system, and more than 12,000 beacons deployed, 800 of them registered customers (shopping centers and shop owners) in Tehran city. Furthermore, more than 400,000 customers' requests are registered per day, in average, which could increase sporadically during special dates. For this reason, reducing the system response time in the activity

Number of Users	Total time	Comm time	Processing time
100	1.168	0.1305	1.0375
200	2.299	0.224	2.075
500	5.589	0.4015	5.1875
1000	11.228	0.853	10.375
4000	42.654	1.154	41.5

TABLE I
TOTAL, COMMUNICATION AND PROCESSING TIMES.

peaks becomes a critical challenge in the AROUND smart recommendation system.

V. AN EDGE-BASED APPROACH FOR AROUND

The current implementation of the AROUND system faces a critical challenge: guaranteeing a low response time for their customers. As the number of users increase the response of the pretrained model becomes unbounded, providing a high delay for the users. In this section, after discussing some alternative options, an edge-based strategy will be presented.

A. Basic Options

The first option is moving the AROUND system smart component to the client side, installing this feature directly into the customers' smartphones and synchronizing periodically the new promotions to that device. In this case, there is no need for a powerful cloud infrastructure and the response time will not be affected neither by the number of concurrent users nor by any eventual network congestion. In fact, the response time will only depend on the end-user smartphone hardware. However, the drawbacks of this strategy are multiple. First, if the smartphone is relatively old or resources are overloaded at the moment they range a specific beacon, the recommendation could take too much time. Second, executing the recommender in the smartphone would overload the customer's personal device, wasting battery and slowing down other apps. And the most important, installing the smart component into the client side would expose security issues, and the system intellectual property and business model could be compromised by rivals.

The second option is increasing the amount of computing resources in the cloud, so that they could absorb the requests load peaks. Some thresholds should be defined in order to keep the response time in an acceptable criteria. This solution may work for the current situation or near future, but the problem is predicting the amount of resources required to guarantee the response time or, in case of over provisioning, an inflated maintenance cost of the system. In addition, this option scales successfully with the number of users, but still exposes high latency when the network is overloaded, as show in the previous section.

In this research work, the option selected for further exploration to reduce the response time latency is leveraging the edge computing capabilities.

B. An Edge-based Strategy

The architectural solution selected for the AROUND system is an edge-based approach. The main idea is moving to the edge the system component that requires immediate response, i.e., the recommender, and keeping in the cloud all global and strategic tasks as backend processes. Each business, or group of businesses in close locations, share some computing facilities for that area. The information they need to manage is the data related to the local advertisement campaigns (a reduced subset of the whole information in the AROUND system) as well as the customer personal information for all clients who have been detected in that area. Note that the first time a customer is detected in a new area, the time to synchronize his information is not critical for a registration stage. Then, when the customer's smartphone range a registered beacon the smart recommendation process is launched locally in a close server. Fig. 5 illustrates this process. The times to send the request and receive the personalized promotion are highly reduced because the server is geographically much closer. In addition, the recommendation process will be more efficient because the problem size is a subset of the whole system.

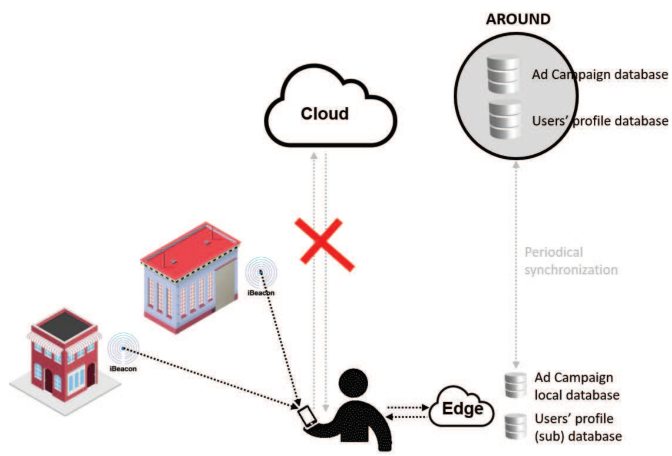


Fig. 5. Edge-based strategy for the AROUND system.

Note that the recommender is based on a categorization of users according to their historical behaviour and their current selected mood. In this process, the first step is categorizing the different types of business into K categories. For each K type category, there are 5 types of mood that have to be taken into account the recommendation system. So 5 * K scenarios should be considered and, for each of them, a different model has to be trained. These models have to be selected cautiously because they have to be merged, in a later process, in the main cloud layer.

The advantages of this strategy are multiple. First, the process is executed locally, so the latency and communication time is much lower. Second, the risk for system overloading is reduced, as the number of customers localized in that area will be much lower than in a global environment. Third, the complexity of the recommendation process is drastically reduced because the number of registers in the databases is a fraction of the whole database. And fourth, by keeping data and processes local, the security risks are minimized. Although this method may look expensive at the point of deployment at first, after a while all of the costs of deployment can be paid off by a growing network of satisfied system users.

VI. Experimental Results

In this section, first we describe the environment considered for experimentation and the methodology used to run the executions. Then, we present some experiments to validate the efficiency of our edge-based strategy with respect to the current implementation of the AROUND system.

A. Methodology

The AROUND system is currently running on the Fandogh cloud provider [20] (an Iranian domestic cloud service, similar to AWS) with 32 nodes 1.8-2.4GHz intel processor, 128 GB DDR4 at 2666 MHz. The system has 6-8 vital services for the main backend tasks and 2 smart services: one for users classification into different clusters and the other for checking the goodness of fit (GoF) model for all classified users in the last 12 hours. If the GoF falls outside the predefined values the system will train a new model (at this point fine tune the same clustering model with new clusters). The main database for recording users data and interaction with the client side is POSTGRESQL 10.16 and the chat data is logged into a Cassandra database. The search option (which is still on staging) is working with Elasticsearch 7.0.

All executions have been done on the real platform. The same trained model (clustering model with customized distance function with size less than 10 KB) has been saved in a .RDS file and loaded within the API once launched. The replication of requests has been done by a for loop in BASH. The worst case scenario has been considered, in which the users who are near a beacon are classified in unbalanced groups and by classifying into the most time consuming cluster. And all the API calls have been done using the identities of users in this exact cluster. The classification has been done by the main trained fine tuned clustering model. Different types of models have been trained by this highly unbalanced data, and the two top models were a multinomial regression and a RBF support vector machine. Although the second one was classifying the users better (99.1% accuracy) the first one is still good (97.2% Accuracy) and is reducing the process time considerably.

In order to validate the efficiency of the edge-based architecture, a new model (multinomial regression model) has been trained with all the 10,000 real users data which ranged some of our registered beacons in a crowded shopping mall in western Tehran. The .RDS file has been deployed with a docker file in a Raspberry Pi at 1.6GHz with Raspbian OS. The main backend service on the centralized server was hard coded to pass these ranged users to the RPI.

B. Edge-based strategy validation

In order to validate the efficiency of our edge-based proposed system, we have run the following experiments. In the first experiment, we compare the performance of the current AROUND system implementation, based on a centralized cloud, with respect to a distributed edge-based architecture. Considering the fact that labels for this particular type of users (at the edge side) are generated with the first trained model, the GoF parameters for this type of analysis have been found to be unstable. As the main model may be very robust for overall performance of the whole data set, this edge side model can be re-trained and fine tuned more frequently. The measured processing time simulated for the edge model at a predefined beacon with the same load of users is shown in Fig. 6.

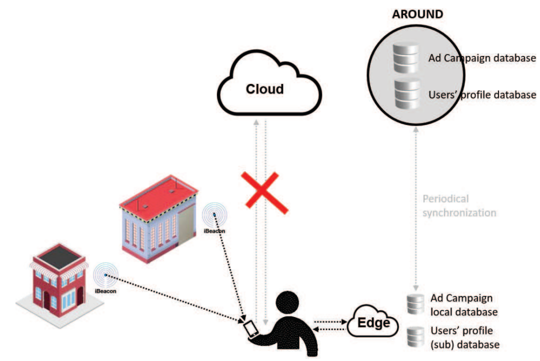


Fig. 6. Processing time in the cloud vs in the edge.

In comparison, the edge-based approach is 62% faster with the same amount of users. Although the .RDS used in the edge side (800KB) is bigger than the main model in the centralized server (10KB) the latter one performs faster as it is not as complex as the centralized model. On the other hand, the number of users distributed at each edge area should be rather smaller than the centralized server. That is the reason why we have chosen a subset (10,000) of the total population of one million users for training a sample of the edge-based model. The detailed times for this experiment can be seen in Table II.

User Counts	Process time for Centralized Service	Process time for Edge Service	%Reduction
100	1.0375	0.382	63.181
200	2.075	0.794	61.735
500	5.1875	2.013	61.195
1000	10.375	3.981	61.629

TABLE II
Processing time in the cloud vs in the edge.

Note that this experiment compares the same number of users in the cloud and in the edge. In a real scenario, where several servers will be distributed through the edge, the number of users per edge area will be much lower than that at the cloud. For instance, assuming 1,000 users sending their requests to the cloud system, the response time would be above 10 seconds. However, in a theoretical scenario with 10 distributed servers, assuming an average of 100 users per server, the response time would drop to 0.38 seconds, reducing the response time in the edge-based strategy in more than 94%.

An alternative for the edge-based strategy is allocating more resources in the cloud, as discussed in the previous section.

In the second experiment, we have measured the response time with our current AROUND system configuration, with a growing number of users ranging from 100 to 4,000, and using two different cluster capacities: 4 nodes and 8 nodes, in order to understand the scalability of the system in the cloud.

Fig. 7 shows the measured response time with both cluster capacities. As can be seen in the figure, the response time is reduced almost linearly with respect to the number of nodes. In fact, there is a 10% overhead compared to a linear scale. We suspect the reason for this overhead is the Kubernetes containers orchestration system.

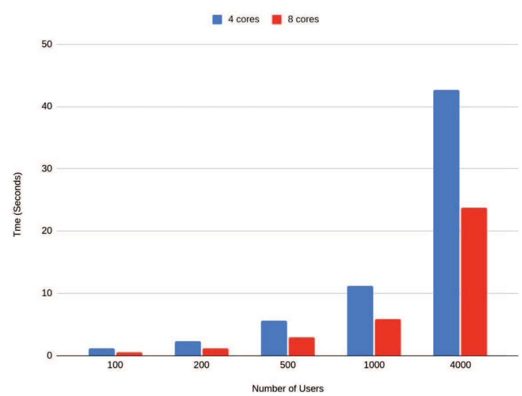


Fig. 7. Processing time in the cloud, with 4 and 8 nodes.

Table III shows the detailed values, and compares them with the edge-based strategy. The response time at the edge is still shorter than that in the cloud (30% approx.), even with the same number of users. Furthermore, the edge-based experiment has been executed on a RaspberryPi at 1.6GHz, compared to the higher performance Intel processor used in the cloud. And if we consider that the number of concurrent requests at the edge will always be lower than that at the cloud, we can conclude that the edge-based strategy is more efficient.

Counts	4 cores	8 cores	Edge-based (Rpi)
100	1.168	0.595	0.382
200	2.299	1.185	0.794
500	5.589	2.975	2.013
1000	11.228	5.863	3.981
4000	42.654	23.724	-

TABLE III
PROCESSING TIME IN THE CLOUD (4 AND 8 NODES) VS IN THE EDGE.

Finally, note that the trained models at the edge are more personalized and have smaller standardized error, and they are expected to be less complex and easier to interpret. For all these reasons, we believe that an edge-based strategy will provide multiple advantages to our AROUND smart advertising system.

VII. CONCLUSIONS

In this paper we have presented the AROUND system, a social media with more than 3 million users and more that 12,000 beacons deployed in Iran and Russia. The core component of this system is a sophisticated smart recommender which takes advantage of a comprehensive set of customers information and broad deployment of a beacons network to trace the users' positioning. After analyzing the performance of the AROUND system in its current implementation, in terms of processing and communication times, we propose an edge-based strategy to improve the response time provided to the system users.

In the experimental section we compare the performance of the current implementation with the edge-based approach, and see that the processing time with the same number of users is 62% faster. However, in an eventual edge-based implementation, the number of users per edge server would be much lower, so this improvement would rise to more than 94%. The advantages of the edge-based strategy are the following. The recommender process is executed locally, so the latency is almost removed from the response time. In addition, the risk for system overloading is reduced, as the number of customers localized in each area will be much lower than in a global environment. Furthermore, the complexity of the recommendation process is drastically reduced because all registers expose a higher locality in the training model. And finally, by keeping data and processes local, the security risks are minimized.

The future line of research will be to implement the edge-based strategy, and adapt the smart recommendation technology to the distributed environment. In this task, several relevant decisions should be taken, such as the management of the trained local subsets, the combination of the distributed learning, as well as the synchronization between the cloud data and the edge servers. Furthermore, the above mentioned results can be optimized even more. By defining different categories of users we can prioritize some requests by assigning fast edge side computing resources or by managing the recommendations before sending them to a load balancer both in the cloud server and in the edge side.

ACKNOWLEDGMENT

This work has been supported by the Spanish Ministry of Science, Innovation and Universities and by the European Regional Development Fund (FEDER) under contract RTI2018-094532-B-I00.

REFERENCES

[1] United Nations, World Urbanization Prospects The 2018 Revision Highlights, Dept. Econ. Social Affairs, United Nations, Tech. Rep. ST/ESA/SER.A/352, 2018.

[2] A. Zanella, et al. *Internet of Things for Smart Cities*. In IEEE Internet of Things Journal, Vol. 1(1), February 2014.

[3] M. M. Rathore, et al. *Exploiting IoT and big data analytics: Defining Smart Digital City using real-time urban data*. In Sustainable Cities and Society, Vol. 40, July 2018.

[4] C. M. R. Haider, A. Iqbal, A. H. Rahman, M. S. Rahman. *An ensemble learning based approach for impression fraud detection in mobile Advertising*. In Journal of Network and Computer Applications, Vol. 112, June 2018.

[5] The AROUND home page. Available at: http://aroundco.com (last accessed: Apr. 2021)

[6] F. Bonomi, R. Milito, J. Zhu, S. Addepalli. *Fog Computing and its Role in the Internet of Things*. In 1st Workshop on Mobile Cloud Computing, Helsinki, Finland, August 2012.
[7] Y. C. Hu, e. al. *Mobile Edge Computing: A key technology towards 5G*. ETSI white paper, September 2015.
[8] K. Y. Kim, B. GyouLee. *Marketing insights for mobile advertising and consumer segmentation in the cloud era: A Q–R hybrid methodology and practices*. In Technological Forecasting and Social Change, Vol. 91, February 2015.
[9] P. C. Verhoef, et al. *Consumer Connectivity in a Complex, Technology-enabled, and Mobile-oriented World with Smart Products*. In Journal of Interactive Marketing, Vol. 40, November 2017.
[10] N. Helberger, et al. *Macro and Exogenous Factors in Computational Advertising: Key Issues and New Research Directions*. In Journal of Advertising, Vol. 49(4), August 2020.
[11] A. Levanov, A. Asmaryan, I. Borovik. *Smart Advertising*. In International Journal of Interdisciplinary Telecommunications and Networking, Vol. 8(4). October 2016.
[12] J. Siryani, B. Tanju, T. J. Eveleigh. *A Machine Learning Decision-Support System Improves the Internet of Things' Smart Meter Operations*. in IEEE Internet of Things Journal, Vol. 4(4), August 2017.
[13] S. Ryu, Y. Park. *How consumers cope with location-based advertising (LBA) and personal information disclosure: The mediating role of persuasion knowledge, perceived benefits and harms, and attitudes toward LBA*. In Computers in Human Behavior, Vol. 112(4), June 2020.
[14] Z. Abbasi-Moud, H. Vahdat-Nejad, J. Sadri. *Tourism recommendation system based on semantic clustering and sentiment analysis*. In Expert Systems with Applications, Vol. 167, April 2021.
[15] L. Quijano-Sánchez, I. Cantador, M. E. Cortés-Cediel, O. Gil. *Recommender systems for smart cities*. In Information Systems, Vol. 92, September 2020.
[16] X. Bai, B. Cambazoglu. *Impact of response latency on sponsored search*. In Information Processing and Management, Vol. 56(1), January 2019.
[17] N. G. Nayebi, A. Abran. *A systematic literature review: Opinion mining studies from mobile app store user reviews*. In Journal of Systems and Software, Vol. 125, March 2017.
[18] N. Ameen, S. Hosany, A. Tarhini. *Consumer interaction with cutting-edge technologies: Implications for future research*. In Computers in Human Behavior, Vol. 120, July 2021.
[19] The Kubernetes home page. Available at: https://kubernetes.io (last accessed: Apr. 2021)
[20] The Fandogh home page (in Persian). Available at: https://www.fandogh.cloud (last accessed: Apr. 2021)