Smart Privacy Protection for Location-Based Services using Queueing Modelling

Chamana H Ram Sai Jeetesh[1,*], *Yerubandi Sai* Sriram[2], *Kavitha* Kayiram[3], *V.N.* Rama Devi[4]

[1, 2] Student, Department of CSE, GRIET, Hyderabad, India
[3] Associate Professor, Department of CSE, GRIET, Hyderabad, India
[4] Associate Professor, Department of Humanities and Basic Sciences, GRIET, Hyderabad, India

Abstract. In the digital era, we are greatly dependent on the popular applications of the Location Based Services (LBS) in our day-to-day activities. The smart phone comes with a variety of applications which acquire the user location and build up user profile like the user activities, hobbies, places of visit, food orders etc. Such sensitive information in the LBS server can pose privacy risk for the user. To safe guard the user from such threat we propose a smart privacy protection technique in this paper that can conceal the user location when using the location based services. We adopt the generation of dummy locations to obfuscate the user original location from the LBS server. The server generates the result set for the dummy user locations. In this work we try to optimize the things at server as well as user ends with two objectives. The first goal is to work towards identifying the overlap in result sets and generate unique and reduced result set with which the communication load on the network can be reduced. The second goal is to prioritize the result set by Queuing model for the result set through which waiting time of the customer can be minimized. We have also illustrated that this model show good performance in terms of the reduced communication load through experimental results.

1 INTRODUCTION

With ever growing digital population, internet connectivity in every device and uninterrupted service the Location Based Services (LBS) [1] are in high use. The advent of smart technology in digital lives have made drastic change in the marketing, services and communication industry. With LBS, we are able to navigate the geography and also avail many other services at the tip of the finger. The smart phones are resource rich and able to offer many of its features based on user location information. The location based services are able to cater to the user requirements like near-by hospital, restaurant, fuel filling station etc. The service providers are continuously monitoring the user requests and their device location for their business as well many third party servers are also gathering user data and them to advertisers. This poses a threat for the user in sharing the location. Often based on the user location and services availed information the user movement profile can be built which poses a privacy risk. Also many untrustworthy service providers will be tracing the profiles for some statistical analysis. Of course location based services are worthy, but at the same time user privacy is also at stake. Without having to compromise on location privacy, still the LBS can be used to the full extent by giving user privacy in sharing the location. Hence, many approaches were proposed in the literature for privacy preserving in LBS.

The geographical boundaries we use to identify a location are the latitude and the longitude coordinates. Similarly, Global Positioning System (GPS) is used to identify a device location and for tracking the movement of the device with the help of network infrastructure. First, the mobile phone is able to detect the latitude and the longitude coordinates of the current location. This is sent to the LBS requesting for service details. The service provider will send a reply based on the details received in the request. This customized service is obtained from the service provider in reply. The service provider has the information regarding the home, office locations, health conditions, political views, banks, and many other details about the person. This can pose a crucial security threat for the person using location based services. Therefore user privacy needs to be protected.

In this Paper, we propose a smart privacy protection technique that can hide the original geographical location of the user from the LBS server. With this scheme, neither the attacker will be able to guess the true location of the user nor will the attacker ever understand about the dummy locations and their behaviour. As the first step towards obfuscation of the user original location, a set of dummy locations around the user location are generated within a given cloaking radius and the region is termed as a cloaking region. These set of

[*] Corresponding author: jeetesh.egnr@gmail.com

dummy locations are sent to the server with the user query. The server processes the query and the result set for all the dummy locations is generated.

These dummy locations are calculated based on user convenience. Later on, according to the user requirements Points of Interest (PoI) are calculated. The nearest PoI are obtained by using searching and sorting algorithms based on their distances to the centroid of dummies to each PoI. Generally the output obtained contains nearest PoI but with repetition which leads to increment of server cost when they are sent to the user. Therefore the repetitions are removed by using distances from the dummy centres and the centroid. Our proposed algorithm uses queueing theory in order to serve the user, thus helps the user to get the service with in less time.

The user queries submitted by the dummy locations to the LBS server are all requesting for a common service almost within the same geography. So, the result dataset whatever the server generates also exhibit certain commonality. Now, the result set may contain some redundant PoI which are eliminated. These unique PoI need to be sent to the dummy users created. As the resultant set is large and the dummy users may not require all the PoI generated by the server, we have proposed a smart privacy protection scheme to optimize the dataset using the concept of queueing theory. We have the large set of PoI and the network channel which need to be used in an optimal manner. Hence we proposed the queueing theory to identify the reduced PoI which are essential for the user while concealing the original user location from the server. We have conducted the experiments in a simulated environment to validate the efficacy of our scheme.

Now, we present the related work in the Second Section. We give the smart privacy protection scheme details in the Third Section along with detailed procedure about the application development. Finally, we present the performance analysis in Section IV and Conclusion in Section V.

2 RELATED WORK

In this Section, we present the recent work done in protecting the privacy of LBS users. The work done so-far in the literature shows that privacy protection can be achieved either by mobile based schemes or trusted third party servers. The approaches used are location obfuscation, encryption, spatial and temporal cloaking or using dummies.

The mobile based schemes [2] use certain computations to conceal the user location before sending the request to the LBS. Hence, these schemes consume precious battery power from the smart phone. The trusted third party server will map the user true location to a dummy location and sends it to the LBS. As this server maintains a database of user locations and the generated dummy locations it is more vulnerable to attacks. This way the user privacy is not guaranteed with the third party server too.

Many of the dummy based anonymization schemes [3] blur the user location by generating dummy locations around the query issuer. The region around the query issuer is termed to be cloaking region. The users within this cloaking region collect their locations and send them as a query rather than sending the exact location. Also it is required to pool the users. But, this spatial cloaking requires users to be available in the cloaking region at the required time.

Obfuscation [4] is to hide away the user location with a near-by famous place. Obviously, the quality of this scheme cannot be ensured as the availability of such near-by locations may not be always possible. Hence, the quality is not guaranteed.

Mobile based schemes generate dummy locations themselves and send it to the LBS. The LBS takes up their individual requests and sends back to the users. Finally the user can decide from the available list based on their distance. But the only concern is the bandwidth and processing time.

To reduce this communication cost, a dummy-based KNN query anonymization method was proposed in [5] using Bayesian model. To protect privacy of the user query, two dummy generation algorithms were used at the client side and a telescopic search method at the server side for the result query. As this scheme uses Bayesian statistics there exists a certain degree of error.

The work in [6] proposed a centroid-based KNN anonymization query method. This scheme concentrates on optimizing the server processing requests by dummy location generation method. The communication cost is reduced but sometimes error may occur in the centroid computations done in this method.

Hence, addressing the drawbacks in these schemes we propose a smart privacy protection scheme for LBS to reduce the communication cost and the error. We present the details in Section III as follows.

3 OUR PROPOSAL

In this Section, we describe in detail our system model, the smart privacy protection scheme and the assumptions we have considered. We intend to develop smart and quick privacy protection method for users of location based services. Our scheme also adopts the framework of Shokri et al. [7].

3.1 System Model

The user submits query for “hospital within one kilometre radius” to the location based server. Typically the request includes the query radius (one kilometre), the point of interest (hospital), and user original location.

We adopt an algorithm to generate dummy locations around the user original location, So that the user original location can be hidden from the LBS server. These dummy locations are generated around the user location and the region which is used to obfuscate the user original location is termed as cloaking region and the radius is cloaking radius. By using circle divided dummy location generation as used in [6] we adopt their algorithm to maintain anonymity in dummy locations also for its advantage of easily available in every smart phone and for its low computation cost.

Now, we present the algorithm to generate dummy locations around the user position in the cloaking region as follows. The user location is given as pos and around the cloaking radius we draw a circle. This circle is divided with angle of 45° into eight parts. Choose a random point on the axis with distance between center of the circle and the point chosen should be less than the cloaking radius. Likewise we get all eight points in the cloaking region of the circle. Now consider every two neighbouring points and obtain the centroids. Repeat this centroid calculation on all other neighbouring points to obtain eight dummy locations in total. These dummy locations in the cloaking region act as eight different users to the LBS server. But, the user query to the LBS server would be the same query. This way the user original location is not traceable by the LBS server or any third party server. For the outside world, it means there are eight users with queries from different locations to the LBS server protecting the user original location privacy. In this method, the cost incurred in attaining location privacy is the communication load on the network and the server. Hence, the goal of this research is to reduce the communication load using the concept of message queueing. Our proposal of smart privacy protection scheme at the LBS server is presented in Section 3.2 as follows.

3.2 Smart Privacy Protection

The user queries submitted by the dummy locations to the LBS server are all requesting for a common service almost within the same geography. So, the result dataset whatever the server generates also exhibit certain commonality. The first goal is to work towards identifying the overlap in result sets and generate unique and reduced result set. The second goal is to prioritize the result set based on the concept of queueing. The server maintains the detailed list of all the Point of Interest (PoI) like hospitals, restaurants, religious places, public places, schools, gyms, super markets, malls etc. along with their geographical location details. When the user queries the LBS server for a PoI within a cloaking radius the server computes the distance between the user location and the list of all the PoI with distance less than the cloaking radius. The PoI within the cloaking region are given as the result set. As the queries are all requesting for the same PoI with in their cloaking region, the results obtained from all the eight queries exhibit certain commonality. Finally the query results are sent over the communication network. Hence we intend to work in the direction of optimizing this result set. We use the concept of queueing to reduce the communication load and also protecting the user privacy. Now, we give a detailed explanation of the result processing at the LBS server in Section 3.3 as follows.

3.3 LBS Server Processing

Our main aim is to identify all the possible locations which are said to be the Point of Interests (PoI) which form the result set for the user. After the generation of dummies it is hard to identify the original user and we consider all the dummies which are generated in the first step as different users. We take each dummy as center and with a query radius we construct circular region in search of PoI around the dummy location as shown in Figure 1.

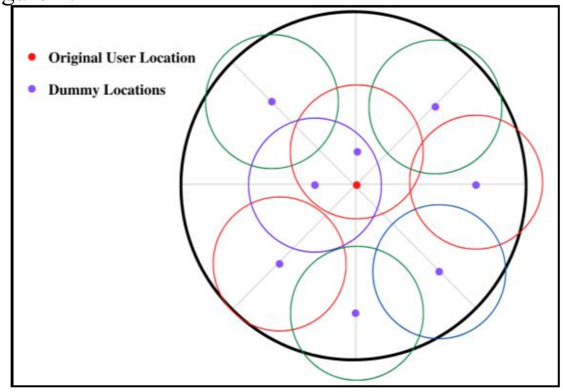


Fig.1. Diagram showing the query region around the dummy locations

From Figure 1, in efforts to optimize the result set, we look at these circular regions generated around the dummy regions. These circular regions are almost inside the required cloaking region. But some of the circles are also found to be outside the cloaking region. This gives us the chance to eliminate those PoI falling outside the cloaking region.

3.3.1 Eliminating the PoI in the no interest zone

It is very clear that the PoI outside the cloaking region limited by its cloaking radius are the PoI which are in the no interest zone and need not be included in the result set. Considering all the PoI will increase the count that is to be computed later by the server in preparing result set. Hence it is necessary to eliminate all the PoI outside the interest zone of the user. But, eliminating them by considering the user original location will disclose the position of user and it will never be anonymous further. So we propose a technique to eliminate the PoI falling in the no interest zone. The Figure 2 shows clearly the PoI falling outside the circular cloaking region.

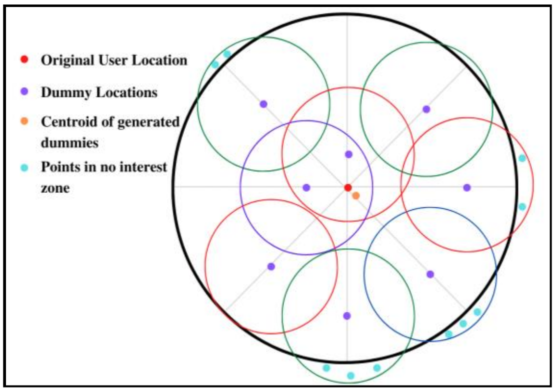


Fig.2. Diagram showing the PoI in the no interest zone.

Taking all the generated dummy locations into consideration, centroid of those dummies will be calculated. Centroid is calculated as follows: Let $x_1,x_2,x_3,x_4,\ldots.x_7,x_8$ and $y_1,y_2,y_3,y_4,\ldots..y_8$ be the x and y coordinates of generated dummies respectively. Therefore, Centroid= $((x_1+\ldots.x_8)/8, (y_1+\ldots.y_8)/8)$.

In this case, the centroid is considered because it is the only point which precisely matches to the original user location. But finding the original user location from the centroid is painstaking. By applying logical and operation to the above two conditions, if either of the condition is not satisfied, PoI in the non interest zone can be eliminated.

In Figure 4, d1 and d2 are the distances from centroid of dummies to the PoI and d3 and d4 are distances from dummies to PoI respectively. Let POI with distances from centroid and dummy d1 and d3 be 'A' and the other one be 'B'. d1 is less than cloaking radius and d2 is greater than cloaking radius , d3 and d4 are less than query radius.In the case of 'A' both the conditions are satisfied, so 'A' falls in interest zone.

But in the case of 'B' the condition 2 is not satisfied, so 'B' falls in non interest zone.

3.3.2 Grouping PoI and eliminating the redundancy

All the PoI which fall inside the circle generated by taking dummy location as center and with query radius will be grouped as result set to the dummy location. But the problem arises when we consider the intersection region of two neighbouring circles as shown in Fig.ure 4. There is a possibility that the circles generated using the two neighbouring dummy locations might intersect and there will be PoI in the intersection region. As we said earlier we are grouping the PoI to dummy locations assigning to their respective circle in which PoI fall. But, PoI in intersection region belongs to both the circles and they have to be grouped to both the dummy locations. But, this kind of grouping increases the number of PoI in the result set and it allows redundancy.

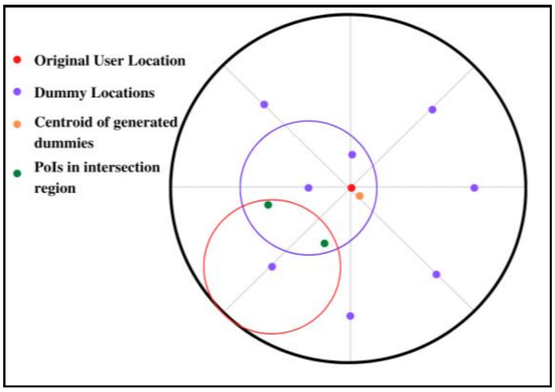

● Original User Location
● Dummy Locations
● Centroid of generated dummies
● PoIs in intersection region

Fig.3. Diagram showing the intersection between two neighbouring circular zone with dummy locations as center.

Eliminating redundancy leads to an optimal result set which helps to reduce the computation time and also the communication load. The PoI in the intersection region will be grouped by calculating the distance between PoI and nearby two dummy location. After calculating the distance, which ever dummy location is close to PoI is identified and then the PoI is added to that group. Thus, the redundancy is eliminated by considering the PoI in intersection region just once while grouping instead of adding them to both the groups. In the Figure 4 shown below d1 and d2 are the distance between PoI and the nearby dummy locations. Hence d2 < d1 the PoI is grouped with the dummy which is closer to it.

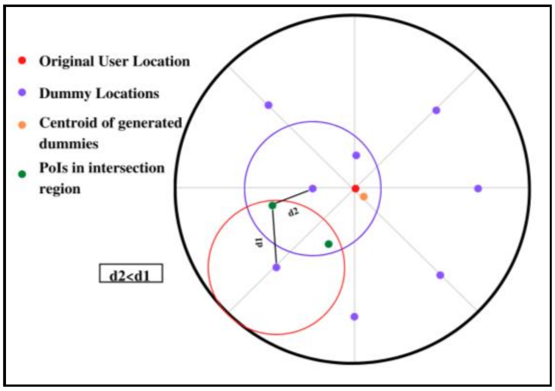

● Original User Location
● Dummy Locations
● Centroid of generated dummies
● PoIs in intersection region
d2
d1
d2<d1

Fig.4. Diagram showing the grouping of PoI and eliminating redundancy.

3.4 Queueing Concept

Now, the results set for all the eight queries is ready after redundancy elimination, which need to be sent to the users. But to hide the user original location from the LBS server, we have generated dummy queries, which increased the communication load on the network. Hence we propose a concept from Queueing theory to reduce the load.For efficient traffic intensity we assumed that mean arrival rate is less than mean service rate. The Queueing theory will let the server recommend the PoI to user which has less waiting time. The waiting time here depends on the size of the result set. Our focus is to optimize the result set and reduce the communication load for a quick and reliable LBS. Hence, we use the concept of Queueing in order to minimize the waiting time of the user.

Let us consider, the dummy locations generated will reach the server with mean arrival rate λ as per Queueing mechanism. The result set generated by the server contains k PoI (say). Now, we need to recommend few PoI to the user based on the concept of Queueing theory explained further in this Section. Here we have multiple recommendations (PoI) obtained for each user, each and every recommendation (PoI) has some respective service rate μij (where μij is the time required for the server to send the information of the jth recommendation of ith user). The service rate is being calculated by considering the size of the PoI which includes the details about the PoI like the public reviews about the PoI, other services offered by the PoI etc. As the details may vary for each PoI, the respective service rates may also vary accordingly which leads to different waiting times. Finally user will receive the PoI in the ascending order of the waiting times. i.e the PoI which has the lowest waiting time will be sent first to the user and later the other PoI with next lowest waiting time .Like this all the other PoI will be shared in a sequential manner.

Assuming that a user places a request for Restaurants which are within a cloaking radius of one kilometer, the dummy locations are generated from the user original location, those dummies are sent to the server with Mean Arrival Rate (λ) equal to 1 micro second (say). Let us also assume that finally there are 11 elements (PoI) for all 8 dummies together in the result set after ensuring that the points are in the interest zone and there is no redundancy in PoI.

For illustration: Out of 11 PoI let there are 2 PoI for dummy user 1, 3 PoI for dummy user 2, 3 PoI for dummy user 3 and rest of the 4 are for dummy user 6.

Consider the respective service rates for each element in the result set as

$(\mu_{11}, \mu_{12}, \mu_{21}, \mu_{22}, \mu_{23}, \mu_{31}, \mu_{32}, \mu_{33}, \mu_{61}, \mu_{62}, \mu_{63}, \mu_{64})$

Now these 11 elements are present in queue with some waiting time, where it can be calculated as follows:

Waiting time$=1/(\mu_{ij}-\lambda)$ $(1\leq i\leq 11)$ (Assuming $\lambda < \mu_{ij}$)

In order to satisfy the users' need in terms of quick response, the PoI will be shared to the user in the ascending order of the waiting times.

This indicates that we have successfully obtained the PoI by pushing minimum load on the network. Now, we validate our proposal with some experiments conducted and results obtained in Section IV as follows.

4. EXPERIMENTAL RESULTS

We have conducted a series of experiments in support of our proposal on smart and quick privacy protection scheme in location based services using a custom built simulation setup developed in Java. To implement our proposal we have used the dataset from zomato.com. All algorithms are run on Windows 10 with a 7.8 GHz Intel Core i7 8th Gen and 8 GB RAM. We summarize the details of our experimental values in Table 1.

Table 1. Experimental Parameters

Parameter	**Setting**
K (number of dummies)	8
Cloaking radius [R]	30(according to the users' requirement)
Query Radius [r]	4, 8, 12, 16, 18,20

Our experiments begin with the redundant PoI elimination. But for evaluation of the result set and to check for its efficiency we are considering another set of PoI with redundancy. By this comparison we are able to validate our technique used in eliminating the redundant PoI from result set and this comparison will help us understand how important it is to eliminate redundancy. For the comparison sake we are considering result set generated using our algorithm and the set which has repeated PoI. We maintain the dummy locations generated as constant during this comparison. Moreover we are generating result set using all the techniques proposed and other set without using the technique to eliminate overlapping (PoI in intersection region of two nearby dummy locations). The optimal case is achieved only after eliminating the overlapping and PoI in no interest zone. To observe the change, many cases are considered in the experiment and many dummy location sets are generated around the user to construct the result set.

The graph in Figure 5 shows the sets on x-axis mean that the dummy location set (8 dummies) and then the PoI in the result set with respect to that dummy location set is observed. In every case we have very few PoI which optimal conditions that to be achieved to save the server time. The optimal is case achieved only after eliminating the overlapping and PoI in no interest zone. To observe the change, many cases are considered to experiment and many dummy location sets are generated around the user to construct the result set. The graph shown below, the sets on x-axis mean that the dummy location set (8 dummies) and then the PoI in the result set with respect to that dummy location set is observed. In every case we have very few PoI which optimal conditions that to be achieved to save the server time.

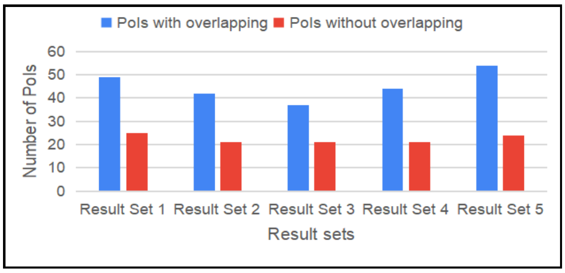


Fig.5. Diagram showing the number of PoI with result set.

The main parameter is query radius, to cover maximum possible number of PoI in the cloaking region. We are performing this experiment to show the importance an optimal query radius to minimize the wastage (wastage means letting the circle generated using dummy locations go out of bounds) and to cover maximum area inside the cloaking circle. Without changing the position of dummy locations we are changing the query radius to observe what is happening to PoI in result set. The graph also gives the comparison between PoI in result set which has overlap with neighbouring circle and after eliminating overlap. The cloaking radius is kept constant so that the dummy location does not change in all cases. In the Figure 6, the x - axis has the ratio of cloaking radius and query radius and y axis has number of PoI obtained.

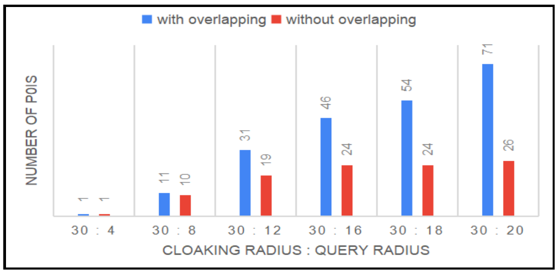


Fig. 6. Diagram showing the PoI with respect to varied query radius

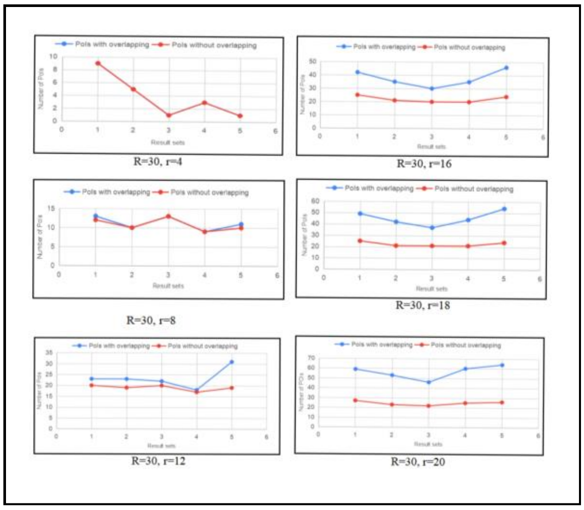
The importance in selecting query radius is, the obvious fact that the circle formed with its center as dummy location has PoI in it. This circle will occupy some area inside the cloaking circle. The area occupied by the circle formed by taking dummy location as center is purely based on the query radius that we are going to set. We get three possible cases as follows.

1. The circle around dummy location may go beyond cloaking region and occupy the area outside the cloaking region which is mere waste. We consider it as wastage because there is a possibility that the PoI outside cloaking circle (PoI in no interest zone) will

fall in dummy circle and thereafter it will consume both time and effort to eliminate the PoI which fall in no interest zone.

2. The dummy circle will occupy only some part of the cloaking circle area and leaving behind the PoI outside. We are expecting this PoI to be in the result set but due to less occupancy of dummy circle we are unable to cover major area of clocking circle. This is a threat to efficiency of result set.
3. Selecting optimal query radius which will cover major area inside the clocking circle and will hardly go beyond the perimeter of clocking circle.

From Figure 7, we get the PoI without overlap. We consider the ratio of cloaking radius with query radius to find the optimal ratio between them, the cloaking radius, R as 30 units and query radius, r varying from 4 to 20 units to generate a graph with cloaking radius /query radius (R : r) on the X- axis and the number of PoI on the Y-axis.



R=30, r=4

R=30, r=16

R=30, r=8

R=30, r=18

R=30, r=12

R=30, r=20

Fig. 7. Graph showing the overlap in the result set.

We now evaluate the application of queueing mechanism in our proposal, the dummy locations generated will go to the server with mean arrival rate λ. After the result set is generated by the server, now the server contains k elements (PoI). So the PoI which has the lowest waiting time is sent to the user and later the other PoI on basis of waiting time. This can be seen in the graph presented in Figure 8. The x-axis of the graph has the PoI and the y-axis is the waiting time in micro seconds. This graph gives the difference in the waiting time of the PoI. With this it is evident that the waiting time differs for the PoI. In order to achieve an optimized result set, we now prioritize the PoI based on their waiting time.

Therefore the highly recommended POI to the user is x9 with waiting time 0.25 micro seconds (approx), and the least recommended PoI is x11 with waiting time 2.99 micro seconds. With this it is obvious, that the communication load cud be reduced by selecting the PoI with minimal waiting time. This aids in minimized network load and optimized result set.

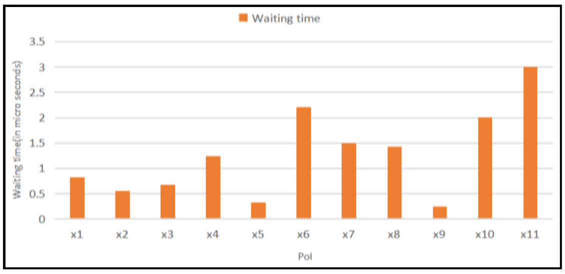


Fig. 8. Graph showing the waiting time

5 CONCLUSION

In this paper, we propose a privacy protection to obfuscate the user location from the server. Our experimental results have proved the efficiency of the scheme. Finally, the proposed procedure requires the server to return only the user needs, thus saving considerable communication cost. The proposed technique can well modify the user experience and guarantee the privacy of the user's location, in the interim, it can save data transmission and improve request accomplishment rate. As future work we intend to investigate the user query privacy and extended datasets.

References

1. Xu, Toby, and Ying Cai. "Feeling-based location privacy protection for location-based services." In Proceedings of the 16th ACM conference on Computer and communications security, pp. 348-357. 2009.
2. Zhao, Huan, Jiaolong Wan, and Zuo Chen. "A novel dummy-based KNN query anonymization method in mobile services." International Journal of Smart Home 10, no. 6 (2016): 137-154.
3. Yi, Xun, Russell Paulet, Elisa Bertino, and Vijay Varadharajan. "Practical k nearest neighbor queries with location privacy." In 2014 IEEE 30th International Conference on Data Engineering, pp. 640-651. IEEE, 2014.
4. Duckham, Matt, and Lars Kulik. "A formal model of obfuscation and negotiation for location privacy." In International conference on pervasive computing, pp. 152-170. Springer, Berlin, Heidelberg, 2005.
5. Zhao, Huan, Jiaolong Wan, and Zuo Chen. "A novel dummy-based KNN query anonymization method in mobile services." International Journal of Smart Home 10, no. 6 (2016): 137-154.
6. Zhao, Huan, and Meng Li. "Centroid-based KNN query in mobile service of LBS." In NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium, pp. 1-6. IEEE, 2018.
7. Niu, Ben, Zhengyan Zhang, Xiaoqing Li, and Hui Li. "Privacy-area aware dummy generation algorithms for location-based services." In 2014 IEEE International Conference on Communications (ICC), pp. 957-962. IEEE, 2014.