Achieving High Privacy Protection in Location-based Recommendation Systems

Tahani Alnazzawi[1], Reem Alotaibi[2], Nermin Hamza[3]
Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia[1, 2, 3]
Faculties of Graduate Studies for Statistical Research, Cairo University, Cairo, Egypt[3]

***Abstract*—In recent years, privacy has become great attention in the research community. In Location-based Recommendation Systems (LbRSs), the user is constrained to build queries depend on his actual position to search for the closest points of interest (POIs). An external attacker can analyze the sent queries or track the actual position of the LbRS user to reveal his\her personal information. Consequently, ensuring high privacy protection (which is including location privacy and query privacy) is a fundamental thing. In this paper, we propose a model that guarantees high privacy protection for LbRS users. The model is work by three components: The first component (selector) uses a new location privacy protection approach, namely, the smart dummy selection (SDS) approach. The SDS approach generates a strong dummy position that has high resistance versus a semantic position attack. The second component (encryptor) uses an encryption-based approach that guarantees a high level of query privacy versus a sampling query attack. The last component (constructor) constructs the protected query that is sent to the LbRS server. Our proposed model is supported by a checkpoint technique to ensure a high availability quality attribute. Our proposed model yields competitive results compared to similar models under various privacy and performance metrics.**

***Keywords*—Recommender models; attacker; privacy protection; dummy; encryption; checkpoint**

I. Introduction

The expression data mining indicates to software tools and mathematical modeling techniques which are applied to detection patterns in data and used to build models [1]. In this concept of recommended applications, the expression data mining is used to describe the set of analysis techniques applied to deduce the rules of recommendation or construct recommendation models from large data groups. Recommender models that integrate data mining techniques build their recommendations based on the knowledge learned from the user's actions and attributes [2].

Fundamentally, recommender systems were categorized into three major types, including Collaborative Filtering (CF), Content-Based (CB) and Hybrid [3]. Later on, combining these based recommender types; novel recommender system types were introduced where location-aware systems are becoming more widespread due to massive usage of smart devices.

In Location-based Recommender Systems (LbRSs), the user requires recommendations for his/her Points of Interest (POIs). To make productive of the required functionality of a recommender system, personal information, along with the current position, is exposed. Fig. 1 shows the traditional way of using LbRSs.

However, disclosing the user's profile, mainly the information of the location, disclose various aspects of one's personal life, which raises many privacy issues because an attacker can deduce sensitive user data by tracing the actual position of the user, such as his/her habits, customs, or religious and political leanings. Therefore, a trade-off exists among recommender systems services usefulness and the privacy of the user.

Research questions: From a privacy perspective, user-based approaches are preferred over server-based approaches because the user has full control in protecting privacy. Based on the description that is presented above, four research questions must be answered in user-based approaches. They are as follows:

1) How we can we select strong dummy locations to achieve high privacy?

2) How can guarantee robustness versus semantic location attack?

3) How can robustness versus sampling query attack be guaranteed to guarantee the privacy of user queries?

4) How can the availability quality attribute be ensured at the user side?

Motivation: The researchers responded to this threat by proposing many privacy protections approaches. Some of them belong to the server-based category and others belong to the user-based category [4, 5]. Many techniques are provided under the server-based category, such as anonymization [6, 7], mix zones [8, 9], and obfuscation [10, 11]. However, the main drawback of the techniques in the server-based category is that the LbRS server itself can be an attacker. Hence, all the LbRS user information and activities are revealed to and attainable by the attacker. This large security gap changed the minds of the researchers, who moved towards the user-based category. In the user-based category, coordinate transform [12], space twist [13], cryptography [14, 15], and dummy [16, 17, 18, 19, 20, 21] techniques are used. In the latter, generating weak dummies enables the attacker to apply advanced inference attacks successfully (position homogeneity attack [22, 23], sampling query attack [24, 25, 26], and semantic position attack [27, 28]). The semantic position attack is considered the most advanced and dangerous among these attacks because the attacker exploits both the semantic meaning of the position and the duration for which the user remains at the position to

deduce personal information about the LbRS user. Therefore, a robust approach versus the semantic position attack is a top requirement. Moreover, in addition to the need to generate strong dummies, disconnection problems may occur any time for any reason, thereby forcing the LbRS user to regenerate the dummies from the beginning, which consumes the power of the LbRS user mobile phones. Dummy generation and short battery lifetime of the mobile device are considered the most important issues in the dummy-based privacy protection approaches.

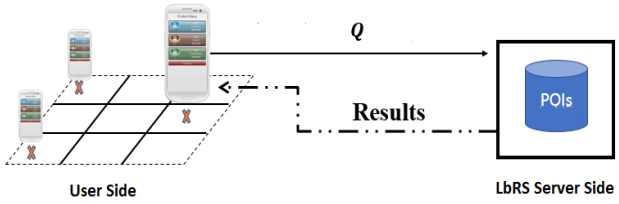


Fig. 1. The Traditional Way of using LbRSs.

To generate strong dummies, ensuring robustness against semantic location attacks, we can select dummy locations that are farthest from the true location of the LbRS user and farthest from each other. To solve the disconnecting problem and ensuring the availability quality attribute, we can use the checkpoint technique to save the last process performed before disconnecting problem happened. Generally, the paper has many contributions they are:

- We propose a new dummy-based privacy protection technique called: The Smart Dummy Selection (SDS) approach. The SDS approach protects the location privacy of the LbRS user by surrounding the actual location by dummy locations.
- To ensure robustness versus semantic location attack, the SDS approach selects the dummy locations such that they are distributed from one another, which weakens attacker ability to know the actual location of the LbRS user between the dummy (fake) locations.
- We strengthen the SDS approach by proposing an encryption-based approach.
- The proposed model is supported through a checkpoint technique for ensuring the availability quality attribute.

The remainder of the paper is arranged as follows: Section II reviews some related works. Section III provides our proposed approach. Security analysis of the proposed approaches is studied in Section IV. Section V shows the results of the experiments and the evaluations in Section VI. Finally, we write the conclusion of the paper to finish it in Section VII.

II. Related Work

Actually, the privacy protection approaches are classified into two main categories and each category has its own techniques, as shown in Fig. 2.

As shown in Fig. 2 above, many techniques are used to protect privacy. Here, we explore the techniques associated with each category.

A. Server-Based Approaches

The recognized feature of this category is that the privacy protection approaches are installed and executed in the LbRS server side. Therefore, the LbRS server is trusted and the privacy can be targeted by an external attacker.

According to [6], k-anonymity provides ensure from multiple k number of users, the concerned user is indistinguishable from k set of users. However, the probability of recognizing a targeted user is 1/k. The problem in this approach was that the user could easily approach to POI through anonymous data. Moreover, k-anonymity approach usually requires the location service as a TTP (trusted third party), which is knowing as exactly location, and work as an anonymizer. In their work [7], the authors provided a personalized K-anonymity approach, where LbRS server works as an anonymizer. This algorithm can be adapting to terms given from the user (i.e., to guarantee the privacy), such that a spatial-temporal mask is used on the location of the user, offering k-anonymity degree.

Likely, a Mix zone, another privacy-preserving approach, was proposed by Zuberi et al. [8]. In this approach, a number of zones are defined where multiple users are positioned in that mix zone. When any user is interest for any recommendation of his/her POI, instead of his/her original location, the whole zone is considered as his/her location for serving. Basically, it refers to a k-anonymization region in which users can change their pseudonyms such that the mapping between their old and new pseudonyms is not revealed. In a mix-zone, a set of k users enter in some order and change pseudonyms, but none leave before all users enter the mix-zone. Inside the mix-zone, the users do not report their locations and they exit the mix-zone in an order different from their order of arrival, thus, providing unlink-ability between their entering and exciting events. In such way, the attacker cannot identify the actual location of any particular user. Although a user's identity and spatial information are indistinguishable, extracting data from a number of zones is an overhead that sometimes causes inefficient results. The mix zone technique is developed by Memon et al. [9]. The core of the evolution concept is to give mix zones extra resistance versus the attackers. To finish this, the researchers considered different types of information which may be used to infer in detail paths such as temporal and geometrical constraints.

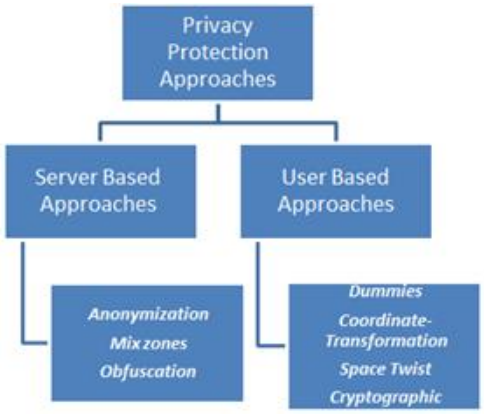


Fig. 2. Classification of privacy protection approaches.

Similarly, Obfuscation was introduced in [10], where reduced information of actual coordinate's location from an obfuscation area is sent to Location Service (LS) and return to the user in a similar way. The fundamental mechanism behind the obfuscation approach is that a query from the user is divided into a number of words where each word is considered as a distinct query. Now attach some dummy terms (anonymous data) along with each query. In response, the user makes the selection of the originally intended answer. Consequently, the precise user location is not shared with clients that maintain the privacy aspects. One problem with spatial obfuscation technique is that the adequate size of the intended obfuscation area can be reduced if an adversary applies background knowledge, last visit, map knowledge etc. in [11]. The obfuscation approach is improved to offer robustness versus semantic location attack.

B. User-Based Approaches

The recognized feature of this category is that the privacy protection approaches are installed and executed in the LbRS user side. Therefore, the LbRS server is untrusted. Even if the server is trusted, privacy can be targeted by an external attacker.

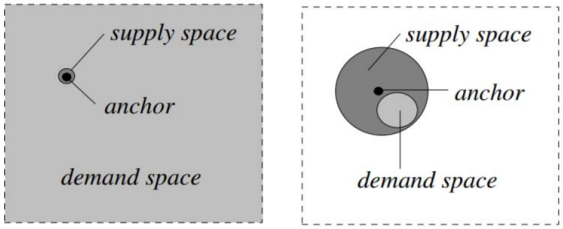
Coordinate transforms technique is presented in [12] to protect privacy. This technique depends on some mathematical operations performed on the coordinates of the true location of the user. The results ensure that the true location of the user will be in another place, which completely differs from the original one.

SpaceTwist was introduced by [13], where the user represents himself/herself as an anchor representative and sends his/her fake location instead of precise information. The further whole communication is based on this anchor location to get any POI as shown in Fig. 3 where supply space cantered at the anchor is the part of space already explored and demand space represents the space to be covered before the client is guaranteed to be able to produce an accurate result. Here the only user knows about these both spaces, but server knows only supply space. At the beginning (Fig. 3(a)), the demand space is set to the domain space, and the supply space is empty. As points are retrieved from the server, the supply space expands. When a retrieved point p is the closest point to the client seen so far, the results are updated, and the demand space shrinks. When the supply space eventually covers the demand space (Fig. 3(b)) it is termed final and the client is guaranteed to produce an accurate result.

Cryptography [14] is another way of protecting user privacy in location-based recommender system. Cryptography approach is based on encryption and decryption mechanism that provide strong privacy. From the user side, the data is firstly encrypted (with a private key depend on cryptography approach) using some algorithms and transferred over the network as shown in Fig. 4 below. The encrypted data is called 'Cyphertext'. On the other side, LS decrypts the data using the same key, which was used during the encryption. However, only LS knows this private key, which totally depends on TTP. Furthermore, as the devices are very smart and requiring results very efficiently, the encryption and decryption processes decrease the efficiency and sometimes cause some swear problems in real road networks.

The cryptography technique is developed by the authors of [15], where secret sharing idea is provided. The key idea is to share the information of the real location among some servers so that the attacker cannot obtain the real location unless collecting the all required information from all servers.

In [16], Yanagisawa et al. gives dummies concept to ensure the LBS user privacy. The conception was that the user generates a number of wrong locations (fake locations), constructing requests of the existing query using both the true location of the user and the dummies, and then submitting all the resulting queries to the LBS server that requesting for the similar POI. Mixing the true location with dummy locations, guarantee the user privacy preserving, where the LBS server cannot determine the true user location from a number of fake dummy locations. Similarly, [17] provides dummies to ensure LBS user privacy protection. It based on chosen the dummy by normalized distance to make the attacker confused and lower his capability to know or deduced important information related to the query generator. Another approach given dummies concept was displayed in [18] called DUMMY-Q, with the different way the dummies are applied to the query instead of the location. Therefore, hide the real query by generating various dummy queries with various features from the same location. To generate stronger dummies, two concepts are considered which is, first, the query form and second, the movement system. Hara et al. [19] improved a dummy algorithm, generate dummies depend on our reality. In this way, they considered the physical imperatives of this present reality. The feature which makes this approach different was that the paths of the fake locations cross the paths of the LBS user real motion. The authors of [20, 21] gave another concept to generate dummy locations, where they depend on selecting the cells that have the same area.



(a) The Beginning. (b) The Ending.

Fig. 3. Demand Space and Supply Space.

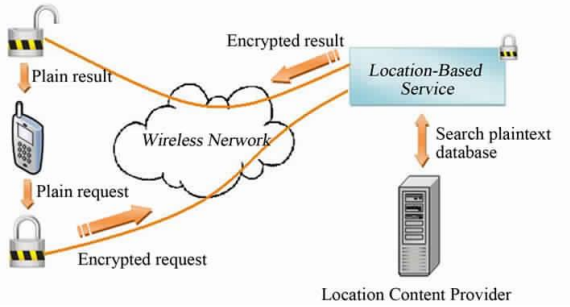


Fig. 4. Cryptography Scheme in LBS System.

III. The Proposed Model

In this section, we provide our idea that guarantees high LbRS users privacy protection. The threat model is presented in A and followed by the identical architecture of our proposed privacy protection model in B. Then, the task of every component located in the model architecture is provided in detail. Our proposed approaches represent the mission of each component. Finally, the architecture detail of the proposed model, which states the interaction among the components, is illustrated by a sequence diagram.

A. Threat Model

To display and identify the threat model, we define the attacker, his/her aim, the kind of the attack, and the skills of the attacker applied to get his aim.

For a given area (RE) split into $(n \times n)$ cells, and many of users (N_{user}) existing over the previous area. The user sends a query of form $Q^t(\langle ID\rangle, \langle Loc\rangle, \langle POI\rangle, \langle Range\rangle)$ to the LbRS server, where t: refers to the moment at which the query is submitted; ID: refers to the identity of the user; Loc: refers to the real position of the user; POI: refers to the queried point of interest; and Range: the range where the queried POI is located or (the search space).

After processing the sent query, the RLbS server returns the results. Since dealing with the LbRS server is inevitable, we consider that the LbRS server is trusted. An external attacker (i.e., a man in the middle) can eavesdrop the communication channel, as shown in Fig. 5.

In Fig. 5, the primary goal of the attacker is gathering personal information about the user to attack his/her privacy. This can be done by monitoring both the sent queries and the retrieved results. In this context, the kind of attack is passive, where the attacker benefits the gathering personal information to construct a malicious profile related to the victim (i.e., the LbRS user). Therefore, no alternation is performed on the sent query or the retrieved results. Specifically, Table I provides the attacker capabilities.

Because of the third capability, the attacker can deduce some user sensitive information based on the area where the user exists. In addition, the attacker can use the duration of time that the user spends in a specific location. Fig. 6 illustrates the way used by the attacker to apply the semantic location attack.

In Fig. 6, the region has three main places (Hospitals-Medical area, Sport clubs-athletic area, and Restaurants-Rest area). The user sends (m) queries $\{(Q_1^{t_1}), (Q_2^{t_2}), (Q_3^{t_3}), (Q_m^{t_m})\}$ to the LbRS server. If the user uses the locations marked by (×) symbols, the attacker can deduce that the user is a patient or an employee in a Hospital even if the user protected his privacy using dummy locations or coordinates transformation approaches. That is because all locations, including the real one, are belonging to the same place (i.e., Hospital or medical area). Moreover, if the attacker analyses the sent queries and discover that the user always asks about the hospitals, he/she will be sure that the user is patient definitely. Furthermore, the attacker can employ the time attached with the queries to estimate the duration that the user stays in this medical area to collect more sensitive information.

B. Our Proposed Model Architecture

The structure of the proposed system composed of trusted LbRS server and several mobile phones linked by a network. The model is work by three components ($selector_D$, $encryptor_{ID}$, and $constructor_Q$) as displayed in Fig. 7.

Table II shows the model components, determines the main job of every component and its installed place.

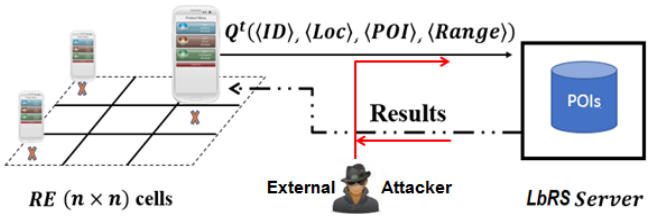


Fig. 5. The Classical Scenario of using LbRS with the Form of the Sent Query.

TABLE. I. The Attacker Capabilities

Capability	Illustration of the Capability
1	track the user location
2	analyze the submitted query after eavesdropping.
3	apply the semantic location attack.

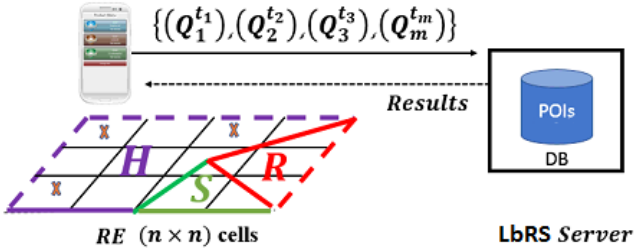


Fig. 6. The Concept of Semantic Location Attack.

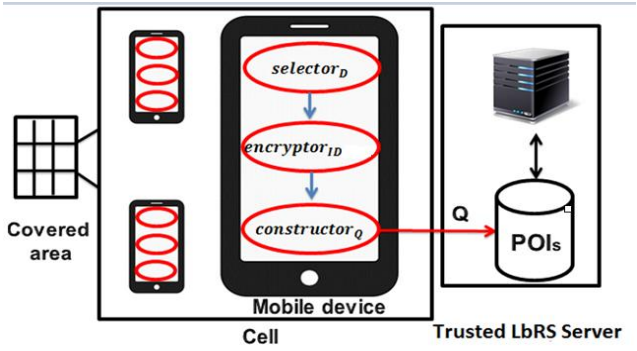


Fig. 7. Our Proposed Model Aachitecture.

TABLE. II. Components

Name	Main Mission	Location
$selector_D$	Dummy locations selection.	Each mobile device.
$encryptor_{ID}$	ID protection.	Each mobile device.
$constructor_Q$	Query construction.	Each mobile device.

The task of each component is combined with and compliant the task of the others. The following clarifies the functions of the components.

C. Roles of Components

Role of the $\text{selector}_{\text{D}}$ component: The eventual aim of this component is to preserve the user location privacy through protecting the location information including the submitted query and the time at which the query is issued (i.e., both information the spatial and the temporal). To end this, this component performs a novel approach called Smart Dummy Selection (SDS) approach as described below.

1) Smart Dummy Selection (SDS) approach: Considering the area (RE) split into a group of cells. Every cell has a query probability. For a particular user located in a cell inside RE, it is a weak solution to randomly select some cells to be the dummy locations. In contrast, it is an efficient solution to select the cells (which will be the dummy locations) that have a similar value probability of the query like the real user cell. Fig. 8 illustrates the idea, where RE is split depending on the coordinates (X, Y).

The method of choosing the locations which have a similar query probability as the true location of the user will confuse the attacker in determining the real location among the dummies. This, in turn, achieves a concept of k-anonymity, where k refers to the number of locations that the attacker cannot recognize the true location of the user between k-1 dummies [31].

Let Q_prob_{i} $(\text{i} = 1, 2, \dots, \text{k})$ refer to the probability that the i^{th} location is the true location. Then $\text{Q\_prob}_{\text{i}} = \frac{\text{qp}_{\text{i}}}{\sum_{\text{j}=1}^{\text{k}} \text{qp}_{\text{i}}}$.

In general, entropy can be defined as uncertainty condition in knowing the true location of a user from all the dummies. In this context, the entropy (ENT) of determining the true location out of the number of dummy locations is defined as [30]:

$$\text{ENT} = -\sum_{\text{i}=1}^{\text{k}} \text{Q\_prob}_{\text{i}} \times \log_2 \times \text{Q\_prob}_{\text{i}} \quad (1)$$

The first factor taken into consideration is to achieve the maximum entropy value in the dummy chosen procedure, which is given by the following formula:

$$\text{Max}\left(-\sum_{\text{i}=1}^{\text{k}} \text{Q\_prob}_{\text{i}} \times \log_2 \times \text{Q\_prob}_{\text{i}}\right) \quad (2)$$

2) The danger of semantic location attack (golden chance for the attacker: Suppose we select the dummy locations by the random way and submit them together with the true location to LBS server. Since the attacker knows the query probabilities of locations in the map, the obtained privacy degree will be down a level. That is because the probability of detecting the real location is $\frac{1}{k}$ which is the theoretic meaning of k-anonymity. Then, the attacker can guess that the real location is the location which has a higher query probability, in contrast, it will exclude all the locations which have low query probability. The gap that is used in selecting the dummy locations, in this case, is illustrated in Fig. 9.

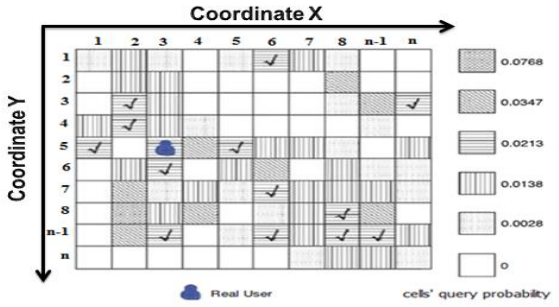


Fig. 8. Dummy Locations Selection in the SDS Approach.

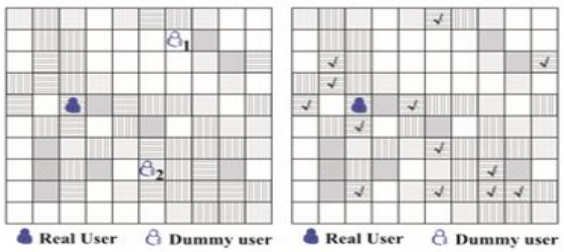


Fig. 9. The Gap in Selecting the Dummy Locations.

Fig. 9(a), since the query probabilities in locations 1, 2 and 3 are more different from each other, the attacker can easily infer the real location from the dummy locations based on the value of the entropy which will drop significantly from $\log_2$ k to $\log_2$ (k−kd), where kd refers to the number of fake locations that the attacker will exclude depend on their minimum query probabilities.

Fig. 9(b) shows the probability of knowing the true location through the server. Since all the candidates have a similar query probability to be aimed as the true location. So, it will be difficult for the attacker to recognize the real location from k locations.

So, our solution improves the user location privacy based on the smart selection of the dummy locations, bearing in mind the attacker may take some advantage of some side-by-side information. So, we select fake dummy locations which have similar query probabilities.

Notice that the temporal information is protected. That is because each dummy location selected by the $\text{selector}_{\text{D}}$ component is used to create a dummy query, which is in turn tightly coupled with the time. Therefore, all the dummy queries will be attached to the same moment at which the real query is issued.

Summarize the following main steps, which are:

1) Choose a suitable degree of k anonymity, which guarantee location privacy without causes system overhead.

2) To achieve maximum entropy, we must have k locations which have the similar probabilities to be aimed as the true location on the server so that we will read all the query probabilities of the cells then we will rank them based on the probabilities values for the inquiry.

3) In the organized set, if we found more than one cells that have the same value of query probability as the true location, we sort half of them before and the other half after the true location.

4) Generate the k cells right before and the k cells right after the real location as 2k candidates.

5) Originate m groups of cells from each with k cells, such that one cell is the true location, and the other (k −1) cells are selected randomly from the 2k candidates.

6) The j^{th} (j ∈ [1, m]) set can be mention as C_j = [c_{j1}, c_{j2},……. , c_{ji},… , c_{jk}]. Depend on the original query probabilities values of the selected locations, the normalized probabilities values of the queries for the included cells can be mentioned as np_{j1}, np_{j2},……, np_{ji},… , np_{jk} and calculated by:

$$np_{ji} = \frac{q_{ji}}{\sum_{l=1}^{k} q_{il}} \ , \ i = 1, 2, \ldots\ldots , k, \quad (3)$$

7) The probabilities sum is 1. The reason behind selecting 2k locations to be candidates of dummy locations is to maximize the anonymity degree, and the number of this group can be changed by the user.

8) Select an optimal list to effectively obtain k-anonymity for the user. We are measuring user's privacy by the entropy-based metric. Specifically, for a selected group C_j , we calculate the entropy by the formula:

$$ENT_j = -\sum_{i=1}^{k} np_{ji} \times \log_2 \times np_{ji} \quad (4)$$

9) In the end, the outputs of the SDS algorithm is the set with the highest value of the entropy:

$$EF_{dum} = \arg\max \ ENT_j \quad (5)$$

Algorithm 1 shows the pseudo code of the proposed SDS approach.

Algorithm 1: Smart Dummy Selection (SDS) algorithm

Input: Q_prob (probability of query for every cell), R_{loc} (the real location of the LbRS user), k (anonymity degree), number of sets m.

Output: EF_{dum}.

1: arrange the cells depend on the value of their query probabilities;

2: choose 2k dummy candidates (k candidates are right before l_{real} and k candidates are right after l_{real} in the sorted set);

3: **for** (j=1; j <=m; j ++)

4: build set Cj which include 1 real and k −1 other cells were chosen randomly from the 2k candidates;

5: calculate the normalized probability np_{ji} of each cell c_{ji} inside the set;

6: $ENT_j = -\sum_{i=1}^{k} np_{ji} \times \log_2 \times np_{ji}$

7: **end for**

8: output $EF_{dum} \leftarrow \arg\max \ ENT_j$;

9: save checkpoint ();

Although the SDS approach can provide a significant level of privacy in entropy domain, as clarified before, it is better to enhance it in terms of protecting user identity by encryption technology.

3) Disconnecting and performance problems: Another task or mission is assigned to the selector_D component, which is dealing with the disconnecting problem that maybe happened to the user mobile device. The disconnecting problem forces the user to go back to the start point of selecting the dummy locations and encrypting the ID element. This, in turn, consumes the power of the device and leads to poor responding performance, where the mobile device of the user suffers from short life battery. To avoid this, the **selector**$_D$ component periodically uses checkpoints to save the performed stage. If the disconnecting problem is happened due to any an error in the user mobile device, the user can go back to the last checkpoint to continue. Notice that this ensures the availability of the system. Fig. 10 illustrates the idea.

Role of the encryptor_{ID} component: After the smart selecting of the dummy locations, these dummies are delivered to the constructor_Q component to create the corresponding queries (dummy queries). since the form of the went query is $Q^t(\langle ID\rangle, \langle Loc\rangle, \langle POI\rangle, \langle Range\rangle)$, we need to protect both the ⟨ID⟩ and the ⟨POI⟩. That is because the attacker can collect some private information from the process of associating the two elements with each other. Breaking this association leads to blocking the attacker from gathering the private data even if eavesdropping is applied on the communication channel. To end this, the encryptor_{ID} component extracts and encrypts the ⟨ID⟩ element. By doing so, the attacker can obtain the following information for example "unknown user is asking for POIs that are located within a specific range", which does not reveal any personal information.

4) Encryption based Approach: To perform the encryption process, we use AES encryption algorithm. Fig. 11 illustrates the extraction and encryption missions of the encryptor_{ID} component.

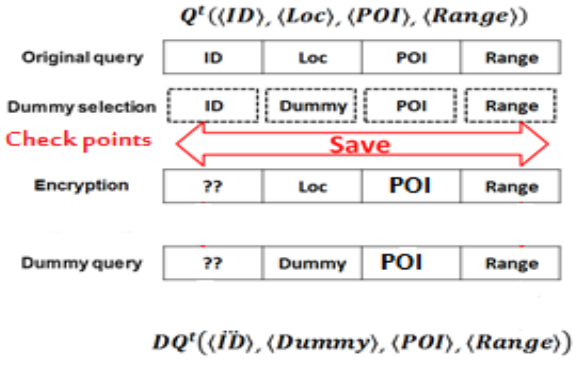


Fig. 10. Checkpoints.

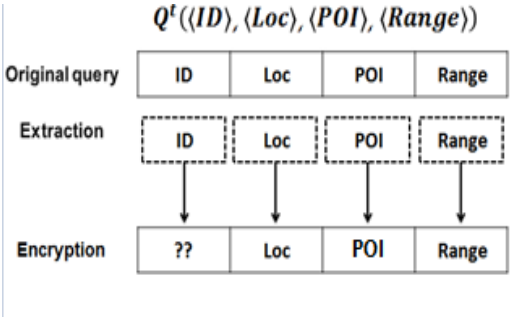


Fig. 11. The Missions of the $\text{Encryptor}_{\text{ID}}$ Component.

5) The danger of sampling query attack: In sampling query attack, the attacker targets the user that is located in an isolated area, focusing on analyzing the sent queries. Since the external attacker has the ability to eavesdrop the communication channel, as defined in the threat model above, he/she can steal the encryption key of the symmetric AES algorithm. This, in turn, facilitates applying the sampling query attack and reflects a gap in the proposed encryption-based approach.

To make a defense against sampling query attack, we need to ensure (1) safely exchanging the encryption key of the AES algorithm; and (2) establishing a secure communication channel between the user and the LbRS server. This can be achieved by using the asymmetric encryption algorithm, such as Diffie-Hellman, as an auxiliary hand.

The Diffie-Hellman key exchange is a secure method for exchanging cryptographic keys. This method allows two parties which have no prior knowledge of each other to establish a shared, secret key, over a public network. This key can then be used to encrypt subsequent communications using symmetric key encryption.

Finally, after safely receiving the session key by the LbRS server side, the two parties start the encryption session through a secure communication channel.

Role of the $\text{constructor}_{\text{Q}}$ component: After encrypting the ⟨ID⟩ element, the mission of the $\text{constructor}_{\text{Q}}$ component is coming. This mission is related to construct the dummy queries using both the dummy locations provided by the $\text{selector}_{\text{D}}$ component and the encrypted (protected) ⟨ID⟩ element by the $\text{encryptor}_{\text{ID}}$ component. The constructed dummy queries have the same form of the original one as shown in Fig. 12, where DQ^{t} refers to the constructed dummy query, $\langle \ddot{ID} \rangle$ refers to the encrypted ⟨ID⟩.

After constructing (k) dummy queries based on the $(k-1)$ dummy locations, the real location, and the encrypted ⟨ID⟩ element, the user sends the all queries to the LbRS server to be manipulated there.

D. Our Proposed Architecture Details

To display the scenario of the collaboration among the components existing in our proposed model we used sequence diagram as shown in Fig. 13.

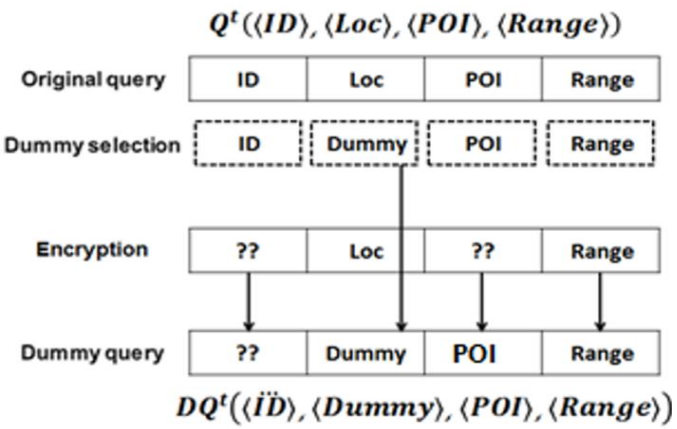


Fig. 12. Construction of Dummy Query.

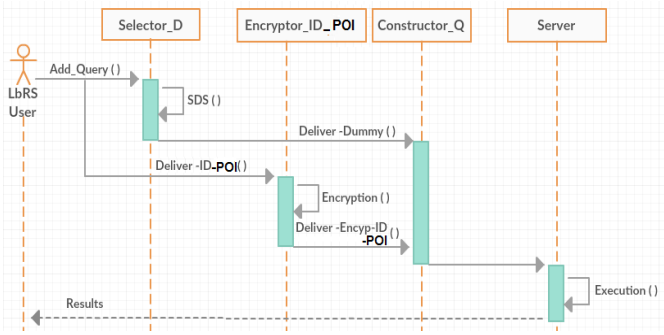


Fig. 13. The Proposed Architecture Details of Sending the Dummy Queries.

IV. Analysis of the Security of our Proposed Model

In this section, we study the robustness of the proposed system versus both the semantic location attack and the sampling query attack. In addition, we study the trial of reversing both the DSD and encryption-based algorithms by the attacker. In this context, we grant that attacker an additional capability, which knows both the SDS and encryption-based algorithms.

In this discussion, we depend on the assumption-evidence strategy to provide proof that the previous two attacks are failed, and the attacker cannot collect personal information about the LbRS user, as described below.

A. Security Analysis of Semantic Location Attack

Assumption 1. We say that our proposed SDS approach is resistant versus the semantic location attack.

Evidence 1. The attacker attempts to deduce the true location of the LbRS user between the dummy locations with the taking in to account the attacker has the following information: (6) the probabilities values of the queries for each cell Q_prob_{i} and (7) all k sent locations, which are mentioned as $\text{l}_1, \text{l}_2, \dots, \text{l}_{\text{k}}$. Let $\text{PG}_{(\text{v})}$ indicate to the probability of the attacker successfully guesses whether an event$v$ occurred. The SDS approach is resistant against the semantic location attack if these two terms are achieved:

$$= PG_{(l_b)} \ \forall \ (0 < a \neq b \leq k) \quad (6)$$

First, because the selected fake locations are having the similar query probabilities Q_prob_i of the actual user's cell (i.e., actual location), the attacker cannot find useful information by using the query probabilities to know the actual locations. Second, as k locations are sent, each a probability of $\left(\frac{1}{k}\right)$ being the actual location. Thus, the first condition is achieved.

However, if we look to the time side, the semantic locations attack is very coupled with time, which means the attacker is given more information. Suppose the attacker knows how many queries that are submitted from a location through a specified time interval. Let $freq\left(Q_{l_a}^{tp}\right)$ and $\text{freq}\left(Q_{l_b}^{tp}\right)$ refer to the frequencies, or how many queries, that are submitted from locations l_a and l_b through the fixed time interval tp . A second condition, which represents robustness versus the semantic location attack, must be achieved when:

$$freq\left(Q_{l_a}^{tp}\right) = freq\left(Q_{l_a}^{tp}\right) \forall\, (0 < a \neq b \leq k) \quad (7)$$

Our proposed system based on the dummy generation process, so all submitted locations (the fake locations plus the user true location) have the same number of queries or the same value of query frequency. That is mean; the attacker only can randomly guess the actual location of the LbRS user because all locations have the same number query frequency. The semantic location attack fails since the two previous conditions are satisfied.

Regarding the SDS approach randomly selects the final dummies (from the candidate list, which is also randomly chosen), even if the attacker understood our proposed approach (SDS), and he tries to overturn the approach; however, this cannot be successful. Due to selecting the dummies by randomization processes, even if the attacker tries to run the proposed SDS approach many times, he will never know the true location of the user between the fake locations.

B. Security Analysis of Sampling Query Attack

We said that the attacker can eavesdrop on the communication channel to analyze the sent query in the description of the threat model above. The sampling query attack requires the original query (decrypting the ID and determining the actual location of the LbRS user) to be obtained. Hence, the attacker must obtain the session key.

Assumption 2. Our proposed encryption-based approach is resistant to query analysis.

Evidence 2. Our proposed encryption-based approach to be resistant to query analysis, the following two security requirements must be satisfied simultaneously:

1) Authentication, which is related to the LbRS server, that means the LbRS server is dealing with the intended user. In other words, if $user_a$ and $user_b$ want to protect queries PQ_a and PQ_b, the LbRS server must safely exchange the session key. (i.e., the LbRS server receives the correct session key that corresponds to the intended user to start the secure communication session).

2) Confidentiality, which is related to the user side, means that no one, except the trusted LbRS server, can obtain the exchanged session key via decryption.

Since the process of exchanging the session key depends on encryption using the Diffie-Hellman key, the previous two security requirements are guaranteed. That is because the received session key is decrypted using the Diffie-Hellman key at the server side. In contrast, if the server tries to decrypt S_{key} using any other key, the decryption process will fail. Since the LbRS server can only decrypt S_{key} using the Diffie-Hellman key, the authentication security requirement is satisfied, and the confidentiality security requirement is satisfied, the sampling query attack fails.

V. The Used Metrics

The metrics that are employed to evaluate our proposed approach are two types of metrics: one for the privacy and another one for the performance.

A. The Metrics of Privacy

Since the main objective of this paper is ensuring high privacy protection for LbRS users, we need to use some metrics related to location privacy and others related to query privacy.

The entropy *ENT*, represented by formula 1 above, is used to evaluate the location privacy of our proposed SDS approach. The entropy privacy metric measures the uncertainty information about the location in LbRSs queries. Consequently, it measures the information an LbRS user can gain from one (or a set) of location update(s) to preserve the privacy. Therefore, if *ENT* value is high, this refers to a higher location privacy protection is achieved, and the lower *ENT* value means a lower location privacy protection. It is worth mentioning that the highest *ENT* value is $log_2(k)$, where this is achieved when all selected dummy locations are exactly treated as the same as the real location.

ENT can be used as a query privacy metric according to [29, 30]. Our encryption-based approach ensures query privacy protection. However, the authors of [29] clearly stated that no query privacy metric can be used for encryption-based approaches. Therefore, we consider the result of encrypting the ID ($\langle \dot{I}\ddot{D} \rangle$) as dummies for the original ID. Here, the attacker tries to link the real location of the LBS user and the ID to infer personal information from the query analysis term. Thus, *ENT* can be used to evaluate our proposed encryption-based approach.

B. The Metrics of Performance

We study two performance metrics: total execution time t_{ext}, and encryption time t_{enc}.

For the time of total execution time, let (t_{ext}) refer the time that is required to extract both the ID at the LbRS user side, (t_{enc}) refer to the time that is required to encrypt the ID of the LbRS user, (t_{sel}) refers the time of selecting the dummy locations, and (t_{rep}) refer to the time that is required to replace the actual location of the LbRS user with a selected dummy location. Then, the total execution time is expressed as follows:

$$t_{ex} = t_{ext} + t_{enc} + t_{sel} + t_{rep} \quad (8)$$

VI. Experimental Results and Evaluations

This section is structured so that the simulation setup is described with configurations. Then, the evaluations and obtained results are represented based on the metrics used above.

A. Simulation Setup

R programming language is used to perform the proposed approaches. To evaluate the performance, we used a Genuine Intel(R) Core (i7) 1.8 GHz PC with 8.00 G RAM that is working Microsoft Windows 10. We download the data set from this link ("https://raw.githubusercontent.com/YunMai-SPS/DA643/master/DA643_final_project/business.csv"). The original dataset consists of 46058 rows and eleven columns: "business_id","name","neighborhood","address","city","state", "postal_code","latitude","longitude","stars","review_count". Due to system limitations, we used a subset of it. The final dataset that we used consists of 8982 rows. Table III lists the parameter settings. Query probability is generated randomly.

B. Evaluations and Discussions

For comparison purposes, we compare our proposed SDS approach with two approaches. One of them is proposed previously called, in this paper, Random Dummy Selection (RDS). RDS selects dummy positions in a random way [20]. The second approach called furthest. The furthest approach selects the dummy locations to depend on the furthest distance among the true location of the user and the chosen dummies without considering the query probability factor. Fig. 14 compares the three approaches under entropy privacy metric, where the K values vary from 12 to 50.

Fig. 14 shows the entropy values according to the increased *k* value of 1 step.

Among the approaches that are represented in Fig. 14, it is evident that the SDS approach overcomes both the random and furthest approaches. That is because of the positive impact of the factor that is considered (in the procedure of selecting the fake locations) in the SDS approach depends on the query probability factor ensures that the attacker cannot determine the true location between the dummy locations since all the locations (i.e., the true location and the fake locations) have the same query probability, which leads to high entropy values. Since both the random and furthest approaches did not deal with query probability as a major factor in the process of selecting dummy locations, this leads to lower entropy values. Compared to the furthest approach, the random approach achieves better privacy protection degree; this is because the random way used to select dummies may match the factor of SDS approach so that some of the selected dummies have a similar query probability as the true location of the user. Fig. 15 compares the three approaches under total execution time in seconds, where K values vary from 20 to 50.

TABLE. III. Configurations

Parameter	Setting
Search Region	5 km
Real User Location	latitude=43.64492 longitude=-79.383333

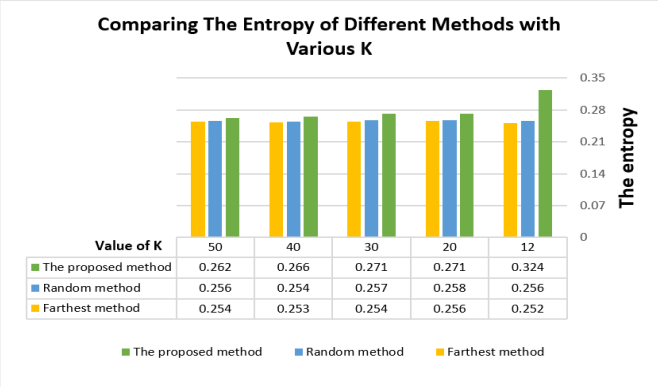


Fig. 14. Entropy vs. k.

Total execution time evaluation: In general, the time needed to execute a privacy protection approach differs from one approach to another due to the time spent to build defenses, as shown in Fig. 15.

It is obvious that the proposed SDS approach performs the worst. That is because (1) the time needed to select the dummy location undergoes the factor of query probability, which in turn consumes more time to end the process of selecting the dummies; and (2) the time needed to encrypt and decrypt the ID of the user. Compared to the random approach, the furthest approach performs less due to the time needed to calculate the distances between the user true location and the selected dummies. The random approach is ranked on the top because it has no factors to be taken into account in the method of chosen dummy locations. As a result, the proposed SDS approach is ranked on the bottom, reflecting the tradeoff among obtained a high privacy degree and execution time, which is a natural result. Fig. 16 compares the proposed approach under total execution time with encryption and without encryption, where K values vary from 20 to 50.

However, although the proposed approach is ranked on the bottom, the difference between the proposed approach under total execution time with encryption and without encryption, where K values vary from 20 to 50, is almost similar with a meager difference.

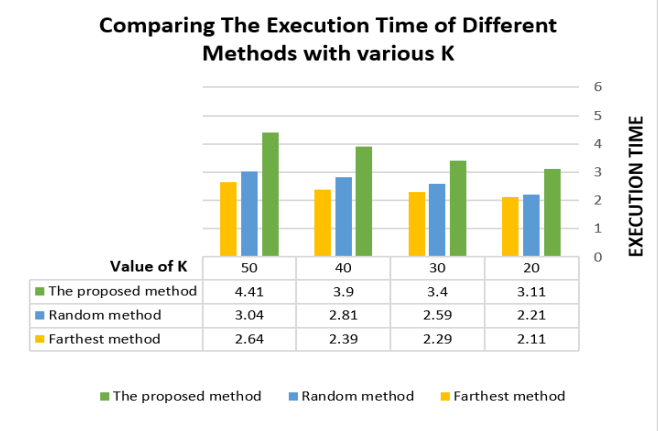


Fig. 15. Comparing the Execution Time of different Methods with Various k.

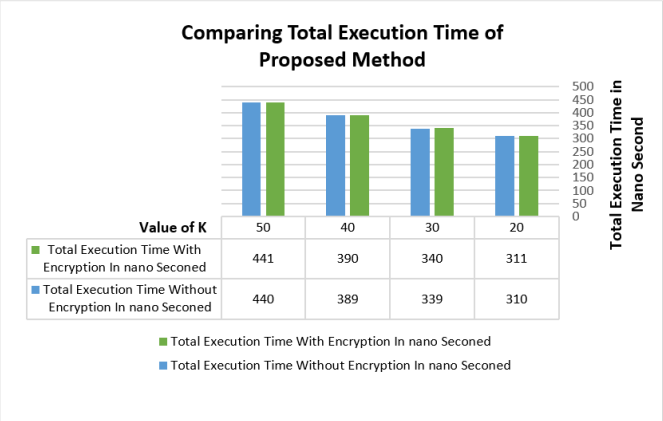


Fig. 16. Construction Time vs. Number of Sent Queries.

VII. Conclusion

Location-based recommendation systems (LbRSs) provide much functionality to users, such as searching for the nearest points of interest (POIs), which facilitates tasks in daily life and saves time. However, the valuable advantages of LbRSs are accompanied by risks since the users are forced to reveal their real locations, which can be exploited by attackers to attack the privacy of the LbRS users. In this aspect of the research field, we contribute by proposing a smart dummy selection (SDS) approach for preserving the location privacy of the LbRS users. To generate strong dummy locations, the SDS approach takes into consideration an essential factor in the process of generating (or selecting) the dummy locations: the query probability of the true location of the LbRS user is equal to each query probability of any selected dummy location. To ensure high privacy protection (i.e., preserving both the privacy of the location and the privacy of the query), we support the proposed SDS approach with an encryption-based approach, which protects the ID. Moreover, the supported SDS approach is strengthened by a checkpoint technique to ensure high availability. Under the threat of a semantic location attack, a sampling query attack, and a mixture of these two attacks, this SDS approach, which is supported by the encryption-based approach, showed higher resistance against such attacks compared to similar approaches.

VIII. Future Work

In future work, we tend to improve this work to deal with other attacks, such as location homogeneity attack. In addition, we will deal with the privacy issue, taking into consideration that the LbRS server itself is the attacker.

References

[1] X. Yang, Y. Guo, Y. Liu, and H. Steck, "A survey of collaborative filtering based social recommender systems," *Comput. Commun.*, vol. 41, pp. 1–10, 2014.

[2] N. A. Albatayneh, K. I. Ghauth, and F.-F. Chua, "Utilizing Learners' Negative Ratings in Semantic Content-based Recommender System for e-Learning Forum," J. Educ. Technol. Soc., vol. 21, no. 1, pp. 112–125, 2018.

[3] E. Aslanian, M. Radmanesh, and M. Jalili, "Hybrid Recommender Systems based on Content Feature Relationship," IEEE Trans. Ind. Informatics, p. 1, 2016.

[4] K. G. Shin, X. Ju, Z. Chen, and X. Hu, "Privacy protection for users of location-based services," IEEE Wirel. Commun., vol. 19, no. 1, pp. 30–39, 2012.

[5] X. Zhang and H. Y. Bae, "Location positioning and privacy preservation methods in location-based service," Int. J. Secur. Its Appl., vol. 9, no. 4, pp. 41–52, 2015.

[6] Z. Montazeri, A. Houmansadr, and H. Pishro-Nik, "Defining perfect location privacy using anonymization," in 2016 Annual Conference on Information Science and Systems (CISS), 2016, pp. 204–209.

[7] D. Li, X. He, L. Cao, and H. Chen, "Permutation anonymization," J. Intell. Inf. Syst., vol. 47, no. 3, pp. 427–445, 2016.

[8] R. S. Zuberi and S. N. Ahmad, "Secure mix-zones for privacy protection of road network location based services users," J. Comput. Networks Commun., vol. 2016, 2016.

[9] I. Memon, Q. A. Arain, M. H. Memon, F. A. Mangi, and R. Akhtar, "Search me if you can: Multiple mix zones with location privacy protection for mapping services," Int. J. Commun. Syst., vol. 30, no. 16, p. e3312, 2017.

[10] R.-H. Hwang, Y.-L. Hsueh, and H.-W. Chung, "A novel time-obfuscated algorithm for trajectory privacy," in 2012 12th International Symposium on Pervasive Systems, Algorithms and Networks, 2012, pp. 208–215.

[11] L. Wang, D. Yang, X. Han, T. Wang, D. Zhang, and X. Ma, "Location privacy-preserving task allocation for mobile crowdsensing with differential geo-obfuscation," in Proceedings of the 26th International Conference on World Wide Web, pp. 627–636, 2017.

[12] A. Gutscher, "Coordinate transformation-a solution for the privacy problem of location based services?," in Proceedings 20th IEEE International Parallel & Distributed Processing Symposium, p. 7–pp., 2006.

[13] M. L. Yiu, C. S. Jensen, J. Møller, and H. Lu, "Design and analysis of a ranking approach to private location-based services," ACM Trans. Database Syst., vol. 36, no. 2, p. 10, 2011.

[14] T. Hashem and L. Kulik, ""Don't trust anyone": Privacy protection for location-based services," Pervasive Mob. Comput., vol. 7, no. 1, pp. 44–59, 2011.

[15] G. F. Marias, C. Delakouridis, L. Kazatzopoulos, and P. Georgiadis, "Location privacy through secret sharing techniques," in Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks, pp. 614–620, 2005.

[16] H. Kido, Y. Yanagisawa, and T. Satoh, "An anonymous communication technique using dummies for location-based services," in ICPS'05. Proceedings. International Conference on Pervasive Services, pp. 88–97, 2005.

[17] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Achieving k-anonymity in privacy-aware location-based services," in IEEE INFOCOM 2014-IEEE Conference on Computer Communications, pp. 754–762, 2014.

[18] A. Pingley, N. Zhang, X. Fu, H.-A. Choi, S. Subramaniam, and W. Zhao, "Protection of query privacy for continuous location based services," in 2011 Proceedings IEEE INFOCOM, pp. 1710–1718, 2011.

[19] T. Hara, A. Suzuki, M. Iwata, Y. Arase, and X. Xie, "Dummy-based user location anonymization under real-world constraints," IEEE Access, vol. 4, pp. 673–687, 2016.

[20] M. S. Alrahhal, M. U. Ashraf, A. Abesen, and S. Arif, "AES-route server model for location based services in road networks," Int. J. Adv. Comput. Sci. Appl., vol. 8, no. 8, pp. 361–368, 2017.

[21] م. ن. مشرف, م. س. العرادي, عناب, صالح, and, "Ensuring Privacy Protection in Location-Based Services Through Integration of Cache and Dummie." جامعة نايف العربية للعلوم الأمنية, 2019.

[22] M. Wernke, P. Skvortsov, F. Dürr, and K. Rothermel, "A classification of location privacy attacks and approaches," Pers. ubiquitous Comput., vol. 18, no. 1, pp. 163–175, 2014.

[23] X. Pan, W. Chen, L. Wu, C. Piao, and Z. Hu, "Protecting personalized privacy against sensitivity homogeneity attacks over road networks in mobile services," Front. Comput. Sci., vol. 10, no. 2, pp. 370–386, 2016.

[24] P. Jagwani and S. Kaushik, “Privacy in location based services: Protection strategies, attack models and open challenges,” in International Conference on Information Science and Applications, pp. 12–21, 2017.

[25] D. Kumar, "A study on Processing of Query with Privacy based Preservation." International Journal of Computer Science IJCSIS, vol.15.10, PP. 88-91, 2017.

[26] C. Lin, G. Wu, and C. W. Yu, “Protecting location privacy and query privacy: a combined clustering approach,” Concurr. Comput. Pract. Exp., vol. 27, no. 12, pp. 3021–3043, 2015.

[27] N. Pelekis and Y. Theodoridis, “Privacy-Aware Mobility Data Exploration,” in Mobility Data Management and Exploration, Springer, pp. 169–185, 2014.

[28] H. Li, H. Zhu, S. Du, X. Liang, and X. S. Shen, “Privacy leakage of location sharing in mobile social networks: Attacks and defense,” IEEE Trans. Dependable Secur. Comput., vol. 15, no. 4, pp. 646–660, 2016.

[29] M. S. Alrahhal, M. Khemakhem, and K. Jambi, “A Survey on Privacy of Location-based Services: Classification, Inference Attacks, and Challenges.,” J. Theor. Appl. Inf. Technol., vol. 95, no. 24, 2017.

[30] I. Wagner and D. Eckhoff, “Technical privacy metrics: a systematic survey,” ACM Comput. Surv., vol. 51, no. 3, p. 57, 2018.

[31] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, “Enhancing privacy through caching in location-based services,” in 2015 IEEE conference on computer communications (INFOCOM), pp. 1017–1025, 2015.