User Motivation Based Privacy Preservation in Location Based Social Networks

Akshita Maradapu Vera Venkata Sai
Department of Computer Science
Georgia State University
Atlanta, USA
amaradapuveravenkat1@student.gsu.edu

Kainan Zhang
Department of Computer Science
Georgia State University
Atlanta, USA
kzhang16@student.gsu.edu

Yingshu Li
Department of Computer Science
Georgia State University
Atlanta, USA
yili@gsu.edu

***Abstract*—Location based Social Networks (LBSNs) have become an integral part of mobile social networks, and with increasing popularity, the use of these networks has become more frequent. With the increasing use of these platforms, a lot of information is leaked, posing serious privacy threats to the users. To handle this, most platforms currently have different privacy settings that are extreme and render the processed check-in data useless to the user as the changes made completely deviates from the user motivation behind the check-in. To this end, we propose a model called User Motivation based Privacy Preservation (UMPP), which provides different privacy policies for different user motivations to retain user motivation for a check-in, which is otherwise lost in most other privacy policies in applications today. To the best of our knowledge, this is the first work that proposes user motivation based privacy policies. We evaluate the performance of our proposed methods on real-world datasets in terms of privacy and information loss.**

***Index Terms*—LBSN, privacy, clustering, user motivation.**

I. Introduction

In recent years, mobile technology has seen a great deal of development on both hardware and communication fronts, and better internet availability has made mobile devices omnipresent. This evolution encouraged several web-based applications to migrate to their mobile versions to provide better reach and experience to their users [1]. Moving to a mobile platform has opened up several opportunities for these applications to provide different location-based services to their user base. For example, Facebook alone has several services that were either improved or introduced after moving to an almost complete mobile operation of their application. One such service is check-ins, which was an already existing feature on Facebook. Now, users can post places they visit on the go and attach pictures or maps pointing to their exact location with their check-in. Another such service is **Facebook Marketplace** [2], which is relatively new to the platform and allows people to use their location for advertising items for sale or discover sales nearby, find apartments, and many more. **Facebook Places** [3], is another new service, which is similar to Foursquare [4], allowing people to use their location to explore their neighborhood. This shows that many social networks today use user's location information in most of their services, thus qualifying them as **Location-based Social Networks (LBSNs)**.

Given the popularity of these networks, it is expected that we have more users signing up for these platforms and taking advantage of their services and features. The most commonly used feature on these platforms is 'posts', also popularly known as 'check-ins'. In these posts, the users share locations that they are visiting with their friends. This is done to get some recommendations about the place or make themselves perceive as interesting, thus helping them make better connections with their social circle [5].

While checking in on LBSNs, the users release a lot of information like the geographical coordinates of the location, the location type (restaurant, stadium, movie theatre), time, if they are already at the place, or are heading towards the location and several other things as shown in Figure 1. Therefore, a simple check-in might release a lot more information than the user has intended to. The released information, combined with other sources, can be used to devise and launch strong inference attacks [6]–[9]. Also, if the user check-ins are frequent, the attacker can collect all such check-ins and launch re-identification attacks (to infer places like Home and Workplace of the user) [10], profile users' daily activities or identify commonly taken routes [11]. Therefore, there is a high privacy risk associated with location check-ins, irrespective of the frequency of check-ins for a particular user.

For example, let us assume that Alice goes to Georgia State University, and on most days, she shares a Facebook post in the morning saying she has reached the university. On another day, she posts about a basketball game she is attending at the university. This particular post also has a few friends tagged in it, and many others were seen making similar posts around the same time. In the former case, the check-ins happen more regularly or consistently in the morning, so one might infer that she is heading to "Work". However, in the latter case, though the check-in location is the same as the former check-in's location, it has a social aspect to it, with tagged friends and many others making similar check-ins simultaneously. The first type of check-in has a more personal intention, like keeping track of her university visits. As this check-in lacks social nature and is more regular, releasing this information over a prolonged period will lead to the attacker inferring

This work was supported in part by the National Science Foundation (NSF) under Grant 1741277 and Grant 1829674.

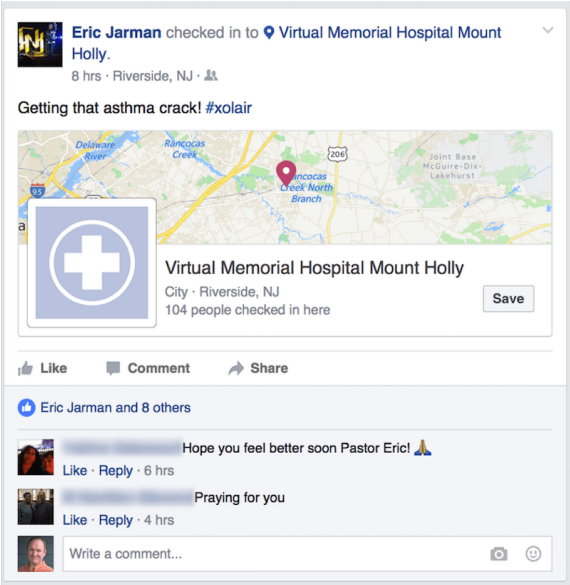


Fig. 1. Example check-in on an LBSN

her activity or the type of location as 'Work'; therefore, it makes more sense to provide stronger privacy for all such check-ins. The second kind of check-in has a social intention behind it, given the number of people making similar check-ins simultaneously, the check-in time, and the tagged friends. In this case, if we provide strong privacy, we will have a higher information loss and may not meet Alice's social needs any longer. Therefore, we can relax the privacy policy (apply a lesser amount of privacy) to retain the 'motivation' behind the check-in.

Several privacy policies are available on social networking applications or proposed as part of research to prevent inference attacks and sensitive information disclosure while releasing the said social network data to third-party applications. The major drawback with the former is that the settings are either so deeply nested in the application that the user might not be able to navigate to them [12] or are complicated for a naive user to understand. Another thing to note is the *motivation* behind the check-ins. Users check-in with some form of social intention, in which case, applying any privacy policy might lead to information (intention) loss and therefore reducing the service quality when the obfuscated information is released. Stemming from the lack of a clear understanding of privacy and privacy settings and the need to satisfy their social needs, the users either opt for complete disclosure (complete location information) or complete non-disclosure (no location information). These extreme settings either have very high privacy risks or very low utility. In privacy-preserving solutions like [13]–[16], all the locations are treated similarly, and the same level of privacy is provided. In these cases, with sufficient knowledge of the user's connections and the network itself, the attacker can back-engineer the policy to obtain the different parameters of the model [17]. Therefore, there is a need for privacy models that take the user's intention (motivation) behind a check-in into consideration to meet both their social and privacy needs and introduce some inconsistency, making it difficult for the attacker to back-engineer the policy.

This paper provides a user motivation-based privacy policy to bridge the gap between user psychology and privacy policies by creating a user motivation based model. The model preserves location check-ins based on the motivation (intention) behind a particular check-in. To our knowledge, this is the first work that has designed a privacy policy centered on user motivation. We consider each check-in individually; therefore, the same user's check-ins might have different policies applied to preserve the information, thus introducing a lot more variation than most other works.

The rest of the paper is organized as follows. In Section II, we review related work on privacy preservation in LBSNs and user motivation based models. The problem definition is provided in Section III. In Section IV, we provide a detailed explanation of our User Motivation Based Privacy Preservation (UMPP) model. In Section V, we cover the experimental results and finally in Section VI, we conclude the paper and provide some future directions.

II. Related Work

A. LBSN privacy

Several research works in recent years have proposed solutions to preserve the private information in LBSN data. This includes preserving the sensitive user profile attributes and location information before releasing the data to third-party applications like recommendation systems. In [18], an algorithm called Equicardinal Clustering was introduced to preserve the user's sensitive information on LBSNs. In this algorithm, all the users are grouped into clusters of same size and, then k-anonymity is applied to each cluster. This solution reduces the information loss significantly compared to other schemes that use traditional clustering algorithms. As the user's neighborhood is not considered for clustering, the privacy provided is not subject to user location, making the solution applicable to both OSNs and LBSNs.

In [19], the authors propose an ML-based model to preserve the user's check-in data. Firstly, the proposed method uses location check-ins and already available user motivation to train a model that predicts future motivation. The users then provide information on the effect of different obfuscation levels on their check-in utility. Based on these responses and the predicted motivation labels, a cost-sensitive decision tree model (J48) is trained to predict the user's perceived privacy level. This solution is the first of its kind as it considers user-specific utility while designing the model that does not use differential privacy. Firstly, it addresses the effect of obfuscation on utility and specifically trains models to predict a privacy level that retains the highest amount of utility. Secondly, designing such intelligent models relieves users from making sensitive and critical privacy decisions.

In [20], the authors propose an obfuscation scheme called **PrivCheck** to preserve the data published on an LBSN. In LBSNs, as users provide access to both historical and future

check-ins, the proposed solution has different obfuscation methods for each scenario. Here, the users indicate the attributes they would like to protect (private data) and the attributes they would like to release (public data). The key idea behind PrivCheck is to preserve the privacy of the user indicated private data and minimize the utility loss under a given data distortion budget.

B. User Motivation

Many privacy-preserving solutions available today focus on only preserving the data, with a utility guarantee only from the application's perspective. While different metrics can justify this perspective, the end user may or may not be satisfied with how the data manipulation tampers with the service he is getting. Therefore, in social networks, considering the user's motivation behind using the application and services may significantly improve user satisfaction. In [21], the authors study and identify the motives and the behaviors in Facebook usage. To this end, they use both the user's personal information metrics and Facebook's usage metrics and train regression models to predict the user's motivation/intention behind using the application.

Studying and understanding user motivation is a psychology problem as it includes understanding several aspects of a user's life. The authors in [22] propose an optimization model called User Check-in Motivation Prediction (UCMP) to learn and predict the motivation behind a check-in by using social psychology based quantifiers. The authors first run a survey on the check-ins made on a Gowalla [23] and gather the true motivation labels from the users of the check-ins. Then their model tries to learn the trends in the motivation labels by minimizing the difference between the true and estimated behavior.

III. Problem Statement

A. Components

Definition 1: **Location based social network (LBSN)**: LBSN can be defined as an undirected graph $G = \{V, E, C\}$, consisting of a set of LBSN users $V = \{v_1, v_2, \ldots, v_n\}$. The friendship relations among the users are represented by the edge set **E**, of the social network, where $e(v_i, v_j) \in E$, indicates that a friendship relation exists between users v_i and v_j, and $v_i, v_j \in V$. **C** reperesents the set of check-ins made by the users on the LBSN.

Definition 2: **check-ins**: A set of check-ins made on a LBSN can be represented as $C = \{c_1, c_2, \ldots, c_j\}$. Each check-in c_i consists of user identifier, location information and the time stamp and can be denoted as $c_i =< v_i, l_i, t_i >$, and all the locations belong to a location universe $L = \{l_1, l_2, \ldots l_p\}$. Each location l_i can be represented as $l_i = \{lid_i, latitude_i, longitude_i, type_i\}]$

B. Computing context features

It is crucial to obtain more information about a check-in and understand it better before designing the privacy policies. We can extract meta attributes, otherwise known as context features, by using both the check-in features and the social network. In this work, the following context features were considered: weekday, time of day, user check-in frequency, location check-in frequency, and co-location.

Weekday: The "weekday" context-feature tells us if a particular check-in has been made during the weekdays (Monday to Friday) or on weekends (Saturday and Sunday). We use the 'timestamp' of the check-in to obtain this information. This is a binary feature, which can be represented as follows:

$$weekday = \begin{cases} 1, & \text{if } day \in \{Saturday, Sunday\} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Time of day: This particular feature tells us if a check-in was made in the morning, afternoon or evening. The check-in's timestamp is used to compute this feature. It can be denoted as follows:

$$time\_of\_day = \begin{cases} 0, & \text{if } Morning \\ 1, & \text{if } Afternoon \\ 2, & \text{otherwise} \end{cases} \tag{2}$$

Location frequency: This particular context feature provides an insight into the activity of a particular location. We compute it by calculating the frequency of a location in all the check-ins made on the system. It can be denoted using *Iversion bracket notation* [24] as follows :

$$location\ frequency(l) = \sum_{i=1}^{|C|} [l_i = l] \tag{3}$$

User frequency: This particular context feature provides us an insight into a particular user's activity. We compute this by calculating the frequency of occurrences of a user v, in all the check-ins made on the system. It can be denoted using *Iversion bracket notation* as :

$$user\ frequency(v) = \sum_{i=1}^{|C|} [v_i = v] \tag{4}$$

where C_i is the set of all the check-ins made by user v_i.

Co-location: Consider two check-ins $c_i =< v_i, l_i, t_i >$ and $c_j =< v_j, l_j, t_j >$. If $l_i = l_j$ (same location), $|t_i - t_j| <= \tau$ (check-ins occuring within a threshold), where τ is the temporal threshold and $e(v_i, v_j)$ exists (v_i and v_j are friends), it is called a co-location. For each check-in, we consider the total number of such co-locations. Algorithm 1, provides the steps for calculating the co-location.

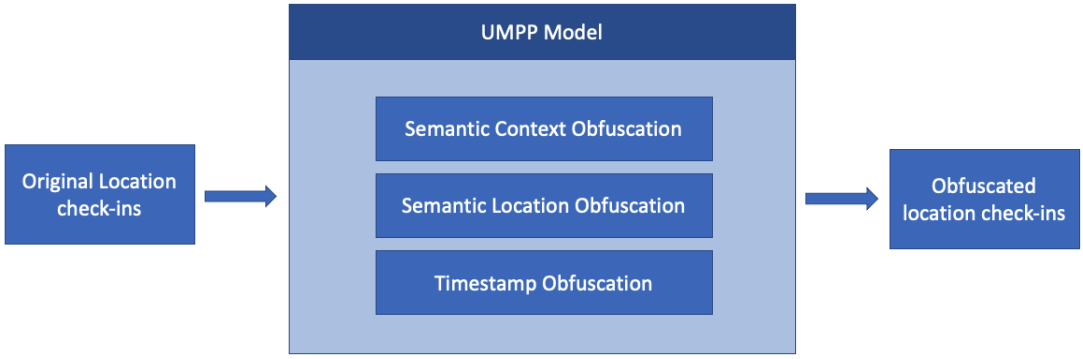


Fig. 2. Structure of UMPP model

Algorithm 1 Computing co-location of the check-ins

Input: $G = (V, E, C)$: the location based social network
C: the set of check-ins
τ: user-defined time difference
Output: the co-locations for all check-ins

1: **for each** check-in $c_i \in C$ **do**
2: Initial $c_i[co-location]$ = 0
3: **for each** check-in $c_j \in C$ $(i \neq j)$ **do**
4: m = $c_i[v_i]$, n = $c_j[v_j]$
5: **if** $|c_i[timestamp] - c_j[timestamp]| \leq \tau$ **and** $c_i[l_i]$ == $c_j[l_j]$ **and** $e_{mn} \in E$ **then**
6: $c_i[co-location]$ = $c_i[co-location]$+1
7: **end if**
8: **end for**
9: Save $c_i[co-location]$ as the co-location for check-in c_i
10: **end for**
11: **return** the co-locations for all check-ins

C. User Motivation

As explained in Section I, every check-in made on an LBSN has an intention or user motivation associated with it. To identify the motivation behind a check-in, we first compute all the context features mentioned in Section. III-B, and then cluster the check-ins based on this data. In [22], it is explained how certain features of the check-in can indicate the intention behind a check-in; therefore, we apply this idea in our labeling. For each cluster, we analyze trends of the computed context features and proceed with labeling the cluster with either of the following two labels:

- Social motivation: A check-in is said to have social motivation if the check-in is made with an intention to communicate a person's whereabouts with others in the network. We apply this label to the check-ins where the location is very active (has high location frequency), the user is active (high user frequency), the check-in has high co-location values (user's friends have also checked in into the same location in the same time), when the check-ins happen in the evenings and/or weekends and finally, when the type of the location is a public place like restaurant, cinema, museum, etc.
- Private motivation: A check-in is said to have a private motivation if the user makes the check-ins for a personal reason. As mentioned in [21], sometimes users check-in only to keep track of their activities. In such cases, the check-ins are more frequent and happen in similar locations over time. We apply this label to check-ins that have less active (less location frequency), the user is active (high user frequency), the check-ins has a less or no co-location and happen on mornings and evenings and/or weekdays and finally when the type of the location indicates home or office.

D. Problem Definition

Given a LBSN network G, as defined in Definition. 1, location check-ins C, as defined in Definition. 2, and user motivation um of the check-in. This paper aims to preserve the privacy of the check-ins in C, based on user motivation um, with the following objectives:

- Minimize re-identification of user motivation behind check-ins C.
- Minimize the information loss for *social check-ins*, while maximizing privacy.

IV. User Motivation based Privacy Preservation (UMPP)

The basic structure of the UMPP model is shown in Figure 2. Each location check-in has three different types of obfuscation applied to it.

- Timestamp obfuscation
- Semantic Location context obfuscation
- Semantic Location obfuscation

Each of these obfuscation techniques effect mutliple context features, thus providing a much better chance against re-identification. Following is the detailed explanation of these obfuscation techniques.

A. Timestamp obfuscation

Timestamp obfuscation is the most commonly used technique used in privacy policies to prevent reidentification and user profiling and tracking. In our model, we adopt a *Reverse kNN* approach to obfuscate the timestamp. As shown in Lines

Algorithm 2 User Motivation based Privacy Preservation

Input: $C = \{c_1, c_2, \ldots, c_i\}$: original check-ins set
$c_i =< v_i, l_i, t_i >$: the original check-in
v_i: user identifer for check-in c_i
$l_i = \{lid_i, latitude_i, longitude_i, type_i\}$: the location information of c_i with id, latitude, longitude, and location type
t_i: the timestamp of c_i
um_i: the user motivation label of c_i

Output: $C' = \{c'_1, c'_2, \ldots, c'_i\}$: obfuscated check-ins set
$c'_i =< v_i, l'_i, t'_i >$: obfuscated check-in
$l'_i = \{lid_i, latitude'_i, longitude'_i, type'_i\}$: obfuscated location information with processed latitude, longitude, and location type
t'_i: obfuscated timestamp

1: **for each** check-in $c_i \in C$ **do**
2: TIMESTAMP OBFUSCATION:
3: Using k-nearest-neighbor algorithm with t_i to obtain a check-ins set $C_j = \{c_{j_1}, c_{j_2}, ..., c_{j_k}\}$ of closest neighbors
4: Randomly select check-in c_{j_i} from C_j
5: t'_i = t_{j_i} (the timestamp of c_{j_i})
6:
7: SEMANTIC LOCATION CONTEXT OBFUSCATION:
8: Generate the lexical location context tree CT_i of l_i over $type_i$
9: **if** um_i == social **then**
10: $type'_i$= 1st level ancestor of $type_i$ in CT_i
11: **else if** um_i == private **then**
12: $type'_i$= 2nd level ancestor of $type_i$ in CT_i
13: **end if**
14:
15: SEMANTIC LOCATION OBFUSCATION:
16: Generate semantic location tree LT_i of l_i over $latitude_i, longitude_i$
17: **if** um_i == social **then**
18: $latitude'_i$= latitude of 1st level semantic location in LT_i
19: $longitude'_i$= longitude of 1st level semantic location in LT_i
20: **else if** um_i == private **then**
21: $latitude'_i$= latitude of 2nd level semantic location in LT_i
22: $longitude'_i$= longitude of 2nd level semantic location in LT_i
23: **end if**
24: **end for**
25: **return** obfuscated check-ins $C' = \{c'_1, c'_2, \ldots, c'_i\}$

2 - 5 of Algorithm 2, we take an individual check-in c_i and generate a nearest neighbor check-in set C_j, containing k nearest check-ins with respect to the timestamp t_i of check-in c_i. We then randomly select one of the nearest neighbor check-ins (c_{j_i}) and use the timestamp of that check-in as the new timestamp t'_i of c_i.

B. Semantic Location Context Obfuscation

A location context is the 'type' or 'nature' of the location like a restaurant, a religious place, cafe, airport, and many more. In most LBSNs, this location type or context information is either directly posted or hidden as metadata when a user checks in; therefore, it can be easily extracted. Given that the context of a location is easily available, even if a privacy policy is applied to the geographical data, one can mine the exact location by checking out how many places within the area share the same context and/or obtain the data from other tagged users. Therefore, it is crucial to consider preserving this information as well in check-ins. In the UMPP model, we preserve this data by applying semantic obfuscation at the lexical level.

For example, let us consider a location which is a "Sushi bar". This is a very specific context for a location. Now if we want to preserve this information, we can move one level up on the lexical tree as shown in Figure. 3, and generalize the location to a "Japanese Restaurant". Furthermore, if we want to preserve more location context information, we generalize it further to "Food/Restaurant". As shown in Lines 7-13 of Algorithm 2, in our model, we move up to different levels in the lexical context tree of location type $type_i$, based on the user motivation um_i of check-in c_i.

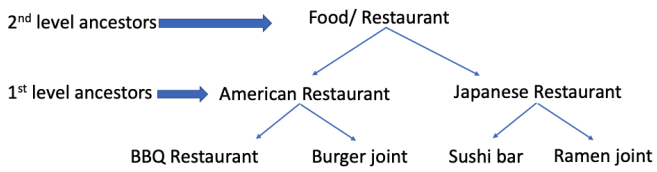


Fig. 3. Lexical tree of location context or types

C. Semantic Location Obfuscation

Semantic Location Obfuscation is a way of generalizing the level of a location's address. As we move up the semantic obfuscation scale, more and more parts of the address are omitted, thus greatly reducing the granularity of the location, and this technique has been found to provide much better preserving preservation compared to simple geographical obfuscation [19]. Figure. 4 shows the levels of semantic information of a sample address format. Therefore, if one needs to preserve more privacy, a higher-level semantic obfuscation can be applied to the location address. In our model, we move up to different levels in the semantics of the address based on the user motivation um_i of check-in c_i, as shown in Lines 15-23 of Algorithm 2, and then we obtain the co-ordinates of that semantic level component (either city or state) as the new co-ordinates of c_i.

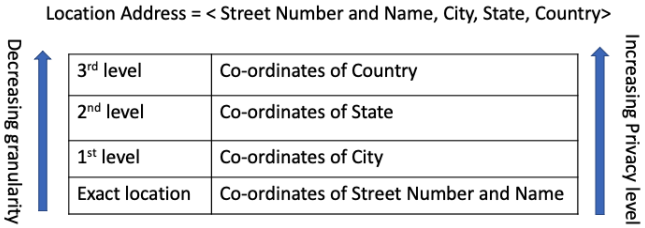


Fig. 4. Semantic levels of location information

V. Performance Validation

A. Datasets

We evaluate our model on two real-world datasets: Gowalla [25] and Brightkite [26]. Both of these are Location-based social networks that allow users to share their locations in the check-ins. In both datasets, we use the check-ins made in the United States over a time frame of 30 days. Table I shows the details of the selected check-ins in both the datatsets.

TABLE I
Dataset Statistics

	Gowalla	Brightkite
Check-ins	35,000	30,000
Users	1900	1794
Locations	498	347

B. Evaluation Metrics

To evaluate the performance of our proposed model, we measure the performance based on two metrics: Re-identification accuracy and Information loss.

1) Re-identification accuracy (RAC): It is essential to note that a check-in's motivation should not be accurately identified after privacy preservation. To measure the **re-identification accuracy**, we first train a classification model that can accurately predict the user motivation. Then, we measure how accurately the classification model can predict the user motivation of the obfuscated check-ins. The lower the re-identification accuracy, the higher is the privacy provided. The re-identification accuracy can be calculated as :

$$RAC = \frac{correctly\ predicted\ motivation\ labels}{Total\ number\ of\ check-ins} \quad (5)$$

2) Information Loss (IL): To calculate the information loss, we use the Average of Sum of Squared Errors (Avg.SSE) metric. The error is calucated between the original checkin c_i and the obfuscated check-in c'_i. Avg.SSE can be calculated as follows:

$$Avg.SSE = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\left(c_{ij} - c'_{ij}\right)^2}{n} \quad (6)$$

where,
n is the number of check-ins, and
m is the number of features in the check-in

C. Results

We evaluate our model's performance against another location check-in privacy preserving model **PrivCheck** [20]. We implement the historical check-in privacy model of PrivCheck for comparison. This model takes user-specified private data and then clusters the check-ins based on the activity/ location type and then applies privacy. To apply the model to the scenario presented in our work, we take all the check-in features as user-specified private features. The results provided are averages over 10 runs of the experiments.

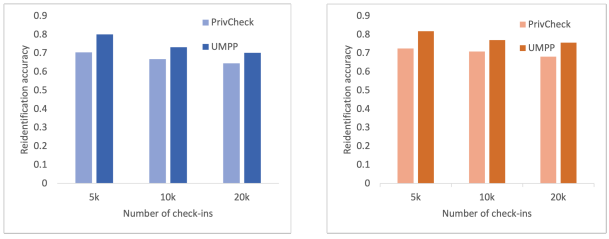
1) Affect on Social Motivation check-ins: The model aims to preserve the social motivation check-ins to some extent while minimizing information loss. Figure. 5 shows that the social motivation check-ins preserved using the UMPP model can be reidentified more than the check-ins preserved using PrivCheck. Furthermore, the information loss of the UMPP model, as shown in Figure. 6, is much lesser than the PrivCheck model. Though the UMPP model provides 9% less privacy over both datasets than PrivCheck, the information loss is almost 20% lower than the PrivCheck model. Therefore, our goal to reduce information loss in "social motivation" check-ins is met by a much higher margin when compared to the baseline, at a very small privacy price.

TABLE II
UMPP on Social Motivation check-ins

	Gowalla		Brightkite	
number of check-ins	RAC	IL (10^2)	RAC	IL (10^2)
5k	0.800	3.248	0.817	2.940
10k	0.732	3.903	0.769	3.438
20k	0.701	4.532	0.756	4.041

TABLE III
PrivCheck on Social Motivation check-ins

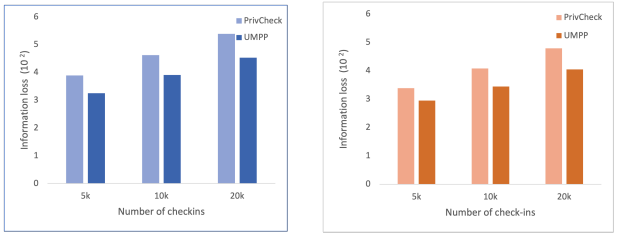
	Gowalla		Brightkite	
number of check-ins	RAC	IL (10^2)	RAC	IL (10^2)
5k	0.703	3.893	0.725	3.386
10k	0.667	4.621	0.708	4.076
20k	0.646	5.384	0.681	4.790



(a) Gowalla (b) Brightkite

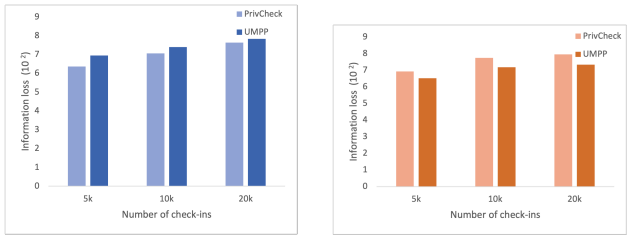
Fig. 5. RAC on Social Motivation check-ins

2) Affect on Private Motivation check-ins: Our model aims to provide more privacy for private motivation check-ins, which is clearly shown in Figure. 7. The UMPP model



(a) Gowalla (b) Brightkite

Fig. 6. IL on Social Motivation check-ins



(a) Gowalla (b) Brightkite

Fig. 8. IL on Private Motivation check-ins

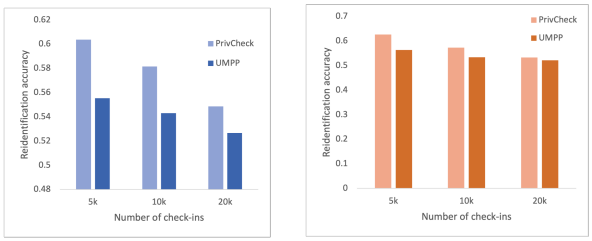
reduces the reidentification of the private motivation labels by 6% compared to the PrivCheck model. If we observe the information loss on the private motivation check-ins in both models on the datasets in Figure. 8, we can see that the information loss is almost the same. Therefore, the UMPP model provides an average of 6% more privacy than the PrivCheck model for the same amount of information loss on both datasets.

TABLE IV
UMPP ON PRIVATE MOTIVATION CHECK-INS

	Gowalla		Brightkite	
number of check-ins	RAC	IL (10^2)	RAC	IL (10^2)
5k	0.555	6.953	0.563	6.516
10k	0.542	7.396	0.533	7.176
20k	0.526	7.827	0.521	7.335

TABLE V
PRIVCHECK ON PRIVATE MOTIVATION CHECK-INS

	Gowalla		Brightkite	
number of check-ins	RAC	IL (10^2)	RAC	IL (10^2)
5k	0.603	6.365	0.626	6.921
10k	0.581	7.065	0.573	7.743
20k	0.548	7.633	0.532	7.951



(a) Gowalla (b) Brightkite

Fig. 7. RAC on Private Motivation check-ins

VI. CONCLUSION AND FUTURE WORK

Mobile devices and their usage have become the norm in today's world. To cater to the users of these mobile devices, many applications in use today have some or all services that LBSNs provide. Therefore, each user is a part of several different LBSNs at any given time. Given the universal nature of LBSNs, the user's data is constantly being used to provide better services. Therefore, information leakage in LBSNs is a major threat to the user. In our paper, we focus on one such service on LBSNs called check-ins. In most current applications, when a user's check-in is preserved, it does not consider the intention or the motivation behind the check-in. This results in losing the meaning of the check-in and, by extension, the utility. Therefore, there is a need to develop privacy policies that take the user's motivation behind a particular check-in into consideration. Our paper proposes a model that divides the check-ins into two categories and applies different privacy policies based on the categories' requirements.

Experimental results show that the model effectively reduces the information loss by about 20% for the 'social motivation' check-ins, at a very small privacy price, as compared to the baseline model. The results also indicate that for the 'private motivation' check-ins, the model provides 6% privacy than the baseline model for almost the same amount of information loss. Therefore, achieving the goals of retaining more information for social check-ins and providing more privacy for private check-ins.

Future work for this paper includes extending the model to more diverse datasets and extracting more features about the check-ins to improve the model further. We also plan on extending the model to online data publishing scenarios as well.

REFERENCES

[1] N. Deshdeep, "App or website? 10 reasons why apps are better." Accessed: May. 24, 2021. [Online]. Available: https://vwo.com/blog/10-reasons-mobile-apps-are-better/.

[2] "Facebook market place." [Online]. Available: https://www.facebook.com/marketplace/.

[3] DCI, "Facebook places for business: Location-based networking at a new level." Accessed: Sep. 07, 2010. [Online]. Available: https://www.dotcominfoway.com/facebook-places-for-business-location-based-networking-at-a-new-level/#gref.

[4] "About foursqaure." [Online]. Available: https://foursquare.com/about.

[5] A. Carman, "Why do you share your location?." Accessed: Dec. 19, 2017. [Online]. Available: https://www.theverge.com/2017/12/19/16792336/location-sharing-apps-privacy-whyd-you-push-that-button-podcast.

[6] A. R. Shahid, N. Pissinou, S. S. Iyengar, and K. Makki, "Check-ins and photos: Spatiotemporal correlation-based location inference attack and defense in location-based social networks," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pp. 1852–1857, 2018.

[7] A. M. V. Venkata Sai and Y. Li, "A survey on privacy issues in mobile social networks," *IEEE Access*, vol. 8, pp. 130906–130921, 2020.

[8] Z. He, Z. Cai, J. Yu, X. Wang, Y. Sun, and Y. Li, "Cost-efficient strategies for restraining rumor spreading in mobile social networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2789–2800, 2016.

[9] M. Siddula, Y. Li, X. Cheng, Z. Tian, and Z. Cai, "Privacy-enhancing preferential lbs query for mobile social network users," *Wireless Communications and Mobile Computing*, vol. 2020, 2020.

[10] T. Anwar, K. Liao, A. Goyal, T. Sellis, A. Kayes, and H. Shen, "Inferring location types with geo-social-temporal pattern mining," *IEEE Access*, vol. 8, pp. 154789–154799, 2020.

[11] A. Noulas, C. Mascolo, and E. Frias-Martinez, "Exploiting foursquare and cellular data to infer user activity in urban environments," in *2013 IEEE 14th International Conference on Mobile Data Management*, vol. 1, pp. 167–176, IEEE, 2013.

[12] M. Madejski, M. Johnson, and S. M. Bellovin, "A study of privacy settings errors in an online social network," in *2012 IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 340–345, IEEE, 2012.

[13] Y. Yao, J. Zhu, J. Liu, and N. N. Xiong, "Verifiable and privacy-preserving check-ins for geo-social networks," *Journal of Internet Technology*, vol. 19, no. 4, pp. 969–980, 2018.

[14] A. R. Shahid, N. Pissinou, S. Iyengar, and K. Makki, "Check-ins and photos: Spatiotemporal correlation-based location inference attack and defense in location-based social networks," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pp. 1852–1857, IEEE, 2018.

[15] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2016.

[16] Z. He, Z. Cai, and J. Yu, "Latent-data privacy preserving with customized data utility for social network data," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 665–673, 2017.

[17] L. Yu, S. M. Motipalli, D. Lee, P. Liu, H. Xu, Q. Liu, J. Tan, and B. Luo, "My friend leaks my privacy: Modeling and analyzing privacy in social networks," in *Proceedings of the 23nd ACM on Symposium on Access Control Models and Technologies*, pp. 93–104, 2018.

[18] M. Siddula, Y. Li, X. Cheng, Z. Tian, and Z. Cai, "Anonymization in online social networks based on enhanced equi-cardinal clustering," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 4, pp. 809–820, 2019.

[19] I. Bilogrevic, K. Huguenin, S. Mihaila, R. Shokri, and J.-P. Hubaux, "Predicting users' motivations behind location check-ins and utility implications of privacy protection mechanisms," in *22nd Network and Distributed System Security Symposium (NDSS)*, 2015.

[20] D. Yang, D. Zhang, B. Qu, and P. Cudré-Mauroux, "Privcheck: Privacy-preserving check-in data publishing for personalized location based services," UbiComp '16, (New York, NY, USA), p. 545–556, Association for Computing Machinery, 2016.

[21] T. Spiliotopoulos and I. Oakley, "Understanding motivations for facebook use: Usage metrics, network structure, and privacy," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 3287–3296, 2013.

[22] F. Wang, G. Wang, and P. S. Yu, "Why checkins: Exploring user motivation on location based social networks," in *2014 IEEE International Conference on Data Mining Workshop*, pp. 27–34, 2014.

[23] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: user movement in location-based social networks," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1082–1090, 2011.

[24] "Inversion bracket notation." [Online]. Available: https://en.wikipedia.org/wiki/Iverson_bracket.

[25] "Gowalla dataset." [Online]. Available: https://snap.stanford.edu/data/loc-Gowalla.html.

[26] "Brighkite dataset." [Online]. Available: https://snap.stanford.edu/data/loc-Brightkite.html.