International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: **10.7321/jscse.v3.n1.2**

A Review of Data Mining Techniques for Privacy in Mobility Data

[1]Sidra Ashraf Khan, [2]Abdul Wahab Muzaffar, [3]Dr. Farooque Azam
[1.] Department of Computer Engineering, College of Electrical & Mechanical Engineering, NUST, Rawalpindi, Pakistan
*[*2.] Department of Computer Engineering, College of Electrical & Mechanical Engineering, NUST, Rawalpindi, Pakistan*
[3.] Department of Computer Engineering, College of Electrical & Mechanical Engineering, NUST, Rawalpindi, Pakistan
Email: [1]sidra.ashraf1@gmail.com, [2]abdul_wahab_muzaffar@yahoo.com

Abstract. **Mobile users' data are becoming increasingly vulnerable due to the many location based services now offered by endless applications in the new mobile app stores. The paper provides an overview of knowledge discovery in databases (KDD) process, and how data mining techniques are used in it. Then, the challenges faced today by user's social-networking habits, which have compromised privacy in an increasingly smart-phone connected world, are addressed. It provides a survey of how research is being carried out in this new and emerging field of knowledge discovery and data mining with respect to data gathered through mobile devices. The different mobility scenarios, possible attack and defense mechanisms for maintaining mobile user privacy is the main focus of this paper.**

Keywords: *data mining, mobility data, anonymization techniques, privacy*

* Corresponding Author:
Abdul Wahab Zaffar,
Faculty of Computer Engineering Department,
Rawalpindi, Pakistan,
Email: wahab_muzaffar2000@yahoo.com Tel:+92-334-5254484

1. Introduction

Mobile telecommunications now provide 4G networking and wifi connectivity options for instant connection to our Wifi and 4G enabled handheld devices. This internet-ready ubiquitous computing has pervaded our society, so that now, the location of mobile users can be continuously sensed and recorded. This has given rise to a set of novel applications that deliver or manage content based on users' location. Examples of such applications are the well-known location based services and, more recently, the location based social networks (e.g., Foursquare, Gowalla) and the participatory sensing systems (e.g., CycleSense) [2]. Competitiveness amongst social networking giants like Facebook and Google plus also add location check-ins of friends as added new features. These emerging trends in applications have given rise to new uses for mobility data.

Besides online applications, advances in database technology and cloud computing now allow mobility data to be available for offline analysis as well. There are many advantages and usefulness of having the ability to store such data. The mining of personal mobility data collected through these applications can produce reliable knowledge of user trajectories to aid traffic engineers, city managers and environmentalists towards decision making in a wide spectrum of tasks, such as urban planning, sustain able mobility, intelligent transportation, and environmental pollution, thereby enhancing the way we live today[2]. In order to harness these abilities through data mining, the data mining community is posed with the challenge of mitigating the user privacy risks involved with these applications. It is only after these risks are addressed, that adoption of such applications will take place.

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

The research objectives of this paper are to survey the privacy techniques being adopted within mobility data in two mobile service scenarios. One is known as snapshot location based services (LBS), where the user sends their current location to an LBS-providing server to gain some service in return at that point in time. The other is known as continuous LBS, where the user queries the server for a service based on their moving trajectory. Users in both scenarios should be given adequate protection in preserving privacy from adversaries that can take advantage of sensitive information regarding users' movement or whereabouts. This paper discusses the various anonymization techniques available in literature to help achieve adequate privacy for the users in both such scenarios.

The paper first gives an introductory overview of knowledge discovery in databases. Then the different mobility architectures are discussed in which the mobility dataset is formed to apply knowledge discovery. The leading techniques are then put forward in Data Privacy and Trajectory Privacy and a discussion of each is followed highlighting its pros and cons.

2. Knowledge Discovery in Databases Process

Knowledge discovery and data mining is a process of seeking patterns among the masses of data that can be stored and organized in modern computer infrastructure [36].Knowledge Discovery in databases (KDD) uses a large dataset, usually stored in a data warehouse or data mart, on which data mining techniques and algorithms are employed to extract some meaningful information. There can be the following five steps in the KDD process -

(1) Selection
(2) Preprocessing
(3) Transformation
(4) Data Mining
(5) Interpretation/Evaluation. [1]

This is not a standard for the KDD Process, and many variants of the process exist where some steps are combined or given alternate names. Figure 1, below shows a variant I created for understanding the steps involved in the process.

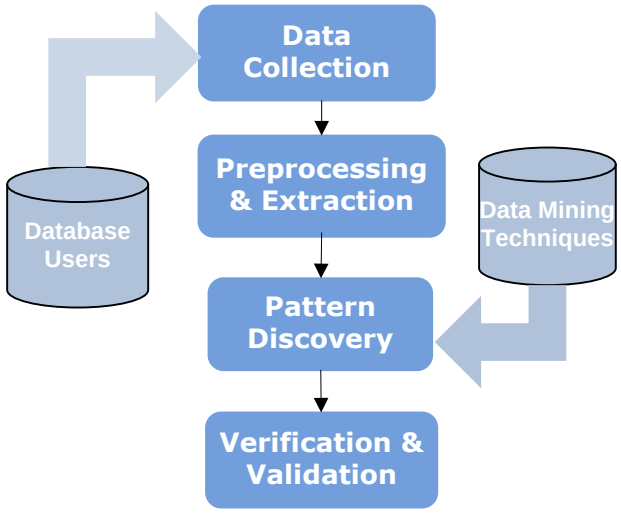

Figure 1 - KDD Process

The Data Collection comes from the centralized or distributed databases (in our case, mobile users) that have shared some information about themselves.

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

The Preprocessing of this data includes a cleansing of data for removing data that may be inconsistent or imprecise. This is a crucial step in extracting any knowledge from the dataset, since noisy or data that has not been cleaned, could lead to false positives or false negatives in the results, thereby making the validation of your results void. Pre-processing is an essential factor in the analysis of the multivariate datasets that exist before data mining. Data mining can only uncover patterns that are actually present in the data; therefore the target dataset has to be large enough to include these patterns while at the same time remaining concise enough so that it can be mined in an acceptable timeframe.

Pattern Discovery is done in the Data mining step. The ultimate goal of data mining is prediction. It is an analytical process designed to explore data in order to find consistent patterns and/or systematic relationships between variables. These patterns and/or relationships are then validated by applying the detected patterns to new subsets of data. There are six basic activities that are common to all Data Mining procedures: [1]

1. Anomaly detection – This detects the outliers, or any deviations from the set of data that could represent unusual data records that could be of interest or it could represent data errors that might need to be investigated further.
2. Classification – Here a known form of pattern is recognized and then generalized to be applied to new data. An example of this would be how an email recognizes the pattern of a spam message and filters it from your inbox.
3. Association rule learning/Dependency modeling – Variable dependencies or relationships are explored to find meaningful associations between them. For example any sales company could gather data on the spending behavior of their customers. Using association rule learning, the company can perform what is known as 'market basket analysis' where they can find out those units that are often purchased simultaneously and then use that knowledge to market those products together.
4. Clustering – In classification we used known pattern forms to classify datasets, here we cluster groups of data that are considered to be "similar" in some way or form, without using known patterns in the data.
5. Summarization – Here a condensed, visualized representation of the data set is provided along with reports generated on it.
6. Regression – A function that can model the data with minimal amount of error is investigated.

The last step in the process of knowledge discovery is Results Validation. The resulting patterns of the data mining (DM) algorithms on the test/training set of data are checked to verify that they also occur in the larger set of data, of which the training set is a subset. It is not necessary for the resulting patterns to be valid on the larger set of data. This is a common phenomenon known as overfitting. This is dealt with by using untrained data, i.e. a subset of the larger dataset in which the DM algorithms do not have any training on. The algorithms apply the learned patterns to this test set and then compare the resulting output with the desired output.

The data mining techniques described above apply to a wide set of data. This wide dataset in the context of this paper refers to mobile users and will be referred to as mobility data.

3. Different Mobility Scenarios

3.1. Privacy in Snapshot LBS and Social Networks

[3] is a survey on the attack scenarios, the offered privacy guarantees, and the data transformation approaches for protecting user privacy are explored in Location based services (LBS). LBS and location based social networks (LBSNs) are popular applications as they enable users to take dynamic, informed decisions on issues like transportation, identification of places of interest, or the opportunity to meet a friend or a colleague in a nearby location. Attacks against user identity, user location, and user query content are discussed and analyzed, with emphasis being placed on the so-

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

called snapshot LBS, in which only one (current) location of the user needs to be reported to the LBS provider to allow the offering of the service.

3.2. Privacy in Continuous LBS, Trajectory Data Publication and Participatory Systems

[4] surveys the state-of-the-art approaches that have been proposed for the offering of privacy in the context of continuous location based services, and trajectory data publication. Unlike snapshot LBS, in continuous LBS a mobile user has to report her location to the service provider in a periodic or on-demand manner, in order to be offered the requested continuous service. Protecting location privacy in continuous LBS is a very challenging problem because adversaries can use the spatio-temporal correlations in the reported location samples for the users to infer their location with higher certainty and breach their anonymity. Continuous LBS can be partitioned into two categories depending on whether they require consistent user identities to offer the service, or do not require user identities. The authors discuss five families of privacy-preserving methods for continuous LBS, namely spatial cloaking, mix-zones, vehicular mix-zones, path confusion and fake location trajectories, and present approaches that fall in each family. Two well studied trajectory anonymization approaches, namely the clustering-based approach and the generalization-based approach, that have been recently proposed for the offering of privacy in trajectory data publishing are considered in the Data Anonymization Techniques section of this paper.

Some interesting privacy challenges that arise in the context of a new breed of applications that are emerging, known as participatory sensing, in which people aim to contribute content via their mobile devices (e.g. of this content could be images or video), to central data servers[21]. This content can be used at a later point to support analytic tasks or other types of data processing. Participatory sensing systems can be either unsolicited (e.g., Flickr or YouTube), where users participate by arbitrarily collecting data, or campaign-based (e.g., CycleSense), when a coordinated effort of the participants is necessary to collect the data that is needed by the data server to support some purpose (e.g., collect traffic information). The authors formally define the problem of Privacy-Aware Participatory Assignment (PAPA) in participatory sensing systems and introduce a privacy-aware framework that enables the participation of users to these services without compromising their location privacy. Although there exist several techniques for the offering of location privacy in conventional LBS, there are certain unique characteristics of participatory campaigns that constitute these approaches inapplicable. The authors propose a method, called Partical-inclusivity and Range independence – PiRi, which solves the PAPA problem and is experimentally verified to be efficient.

Unlike the case of location based services and participatory sensing systems, where privacy has to be offered in an on-line, potentially dynamic, and service-centric manner, in anonymous personal mobility data publishing, the goal is to construct and publish a dataset that effectively maintains most of the utility of the original mobility data, while it effectively protects the personal and sensitive information of the users from potential attackers [20]. The approaches that have been proposed so far assume that a trusted entity collects trajectories representing the movement for a large number of users. This data has to be shared with a set of potentially untrustworthy entities for data analysis purposes, e.g. to enable traffic optimization research. To accomplish this goal, the trusted entity has to anonymize the mobility data so that no privacy breach, based on the assumed models of adversarial attacks, may occur when the data is published. The authors categorize adversary knowledge into two types, namely location based knowledge, and mobility pattern-based knowledge. For each category, they explain the considered privacy model and elaborate on the anonymization methodologies that have been proposed to offer user privacy.

4. LBS Communication Architectures

The scenarios described above function in certain communication architectures. LBS communication architectures are based on one of the three formats described below.

4.1. Communication Architecture 1. The Trusted Server

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

Mobile users communicate with a trusted server that acts as an anonymizer. It performs data transformation on the user query and performs anonymization techniques to make sure that the user's identity is not compromised during communication with the LBS provider. This architecture is widely adopted since it provides computational speed up by having the trusted server perform bulk anonymization of multiple user requests by putting them in equivalence classes having some collective common property, and then applying anonymization techniques. A modified architecture appears in [7, 8] where the trusted server does not communicate itself with the LBS provider, but anonymizes the request and returns it to the client who handles all communication with the LBS provider. The authors assume that given the increase of the computational power in handheld devices it might be possible for the anonymization to fully take part in the client side.

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

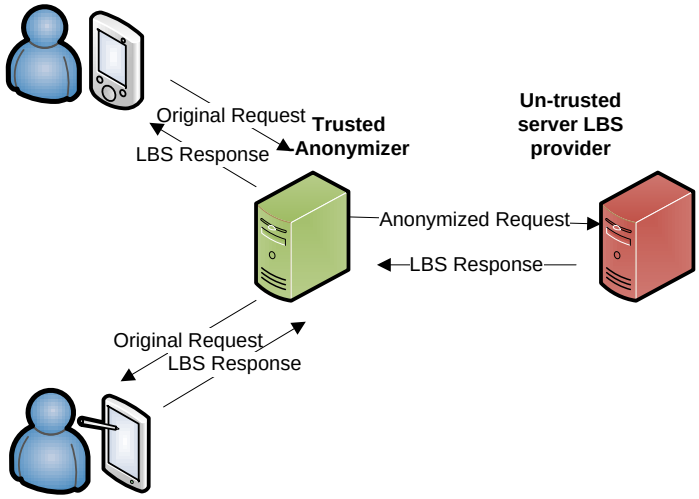

Figure 2. Trusted Server

4.2. Communication Architecture 2. LBS provider acts as an Un-trusted Server

Here the users communicate directly with the LBS provider. The user is responsible for obfuscating sensitive information. One way of doing so is by employing cryptographic protocols, for e.g. Private Information Retrieval protocol [6, 9] for the communication between the user and the LBS provider.

The content of the user query remains hidden in such architecture, however, the location and the existence of a user request to the server remains known. Other works in literature suggest non cryptographic methods that ensure the exact user location still remain hidden. In [12] the user sends a fake location to the LBS provider, which lies close to his actual location, in order to get his nearest neighbors. In [10, 11] the user replaces in the request his real location with a region that contains it. In all cases the protocols guarantee that the exact user location is hidden from the LBS provider.

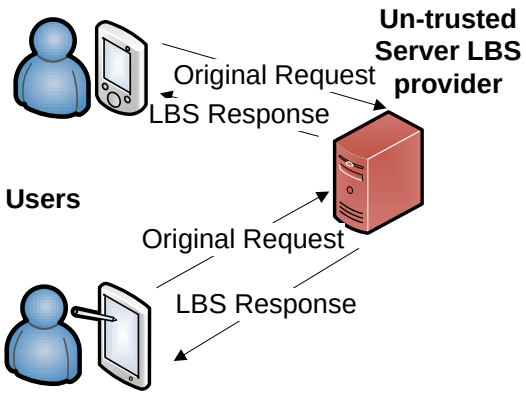

Figure 3. LBS – Un-trusted Server Communication Architecture

4.3. Communication Architecture 3. Direct Communication

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

There are two versions of this scenario in the literature: a) one where the user does not trust the LBS provider, nor the other users and b) and one where users trust each other but do not trust the

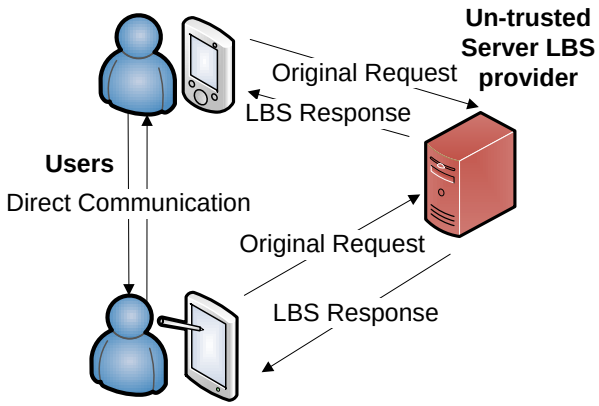

Figure 4. LBS - Direct Communication Architecture

LBS provider. The most common application case that adopts the former version of this scenario is that of proximity based services where a service depends on nearby users [10]. In [13] the authors propose a cryptographic protocol for per-forming the proximity test and in [11, 10] the users send an obfuscated version of their location, so they can get approximate answers about to the proximity test. In [14] the user trusts all other users and communicates with them in a structured peer-to-peer network. The adversary in this scenario is the LBS provider and the users collaborate in order to protect themselves against identity disclosure. Users need a trust certification to participate in the network, but they are all considered trustworthy. They are organized in clusters, and each cluster has a leader. Leaders are organized recursively to other clusters with new leaders.

The network architecture allows each user to acquire information from nearby users in order to anonymize his request before sending it to the LBS provider. It is one of the few examples where users can achieve protection against identity and creation of equivalence classes without the need of an anonymizer.

4.4. Communication Architecture 4 – Mobile Ad-Hoc Networks (MANETs)

MANETs are self-organized networks of mobile users who communicate in order to exchange information. In this setting there is no LBS provider or anonymizer and queries are answered by other members of the network. Every user is considered un-trusted and each user is responsible for anonymizing its own messages. In this architecture it is very hard to create equivalence classes and provide protection against identity disclosure. In ad hoc networks, the main issue about designing of protocols is quality of service, so that in wireless sensor networks the main constraint in designing protocols is limited energy of sensors [37]. Hence privacy is a less considered factor here.

5. Privacy Defense Mechanisms – Data Anonymization Techniques

Within the architectures discussed above, the user sensitive data has to be protected. This is done by transforming the data that the user requests, along with the location and/or identity of the user.

The various techniques employed take into account not only the nature of the LBS, whether it is snapshot or continuous, but also whether they are real-time data or offline data to be published. The techniques most commonly used for anonymization that are discussed in this paper are as follows –

Table 1. Privacy Techniques

Data Privacy	Trajectory Privacy

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

k- anonymity ℓ - diversity *t* -closeness	Spatial Cloaking Mix Zones Dummy Trajectories Path Confusion

The first three deal with anonymizing data of the user, for example identity or sensitive information that is static. The last four are more concerned with protecting user movement or trajectories from adversaries. Protection in highly dynamic data related to the user, mostly in continuous LBS scenarios.

5.1. *k*-Anonymity

In *k*-Anonymity, we take a set of identifying attributes that when put together can uniquely identify a record within the database. We transform these set of attributes by either suppressing them or substituting them with a generalized version until every row is identical with at least *k*-1 other rows so that we have a *k*-anonymous database. This kind of privacy defense mechanism protects individuals against identity disclosure. With *k*-anonymity, you cannot differentiate a record from a set of records with respect to their quasi-identifiers [5]. Quasi-identifiers are those attributes that when combined together, for example like a composite candidate key, can be used to identify individuals. [15] tells us that eighty seven percent of the US population can be uniquely identified by gender, date of birth, and five-digit zip code. These three attributes would then constitute the "quasi-identifier". Datasets are "*k*-anonymous" when you are unable to distinguish a record from *k*-1 others, for any given quasi-identifier. For example, they replace the quasi-identifiers of all records in each group with a common value, e.g., all different salaries are replaced with the average salary of the group. We term these groups with indistinguishable quasi-identifiers as equivalence classes. An example of how this works is shown in Table2 and Table3, where the dataset in Table2 is made 2-Anonymous by suppressing attributes to make 2 rows identical.

Table 2. Original Table

First	Last	Age	Race
Harry	Stone	34	Afr-Am
John	Reyser	36	Caus
Beatrice	Stone	34	Afr-Am
John	Delgado	22	Hisp

Table 3. 2-Anonymous Version of Table 2

First	Last	Age	Race
*	Stone	34	Afr-Am
John	*	*	*
*	Stone	34	Afr-Am
John	*	*	*

Here, we have suppressed identifying attributes with *'s, which represent some quasi-identifiers. To make sure we have k-anonymity with minimum cost, the least number of cells will be suppressed to gain anonymization. If we were to use Generalization-based *k*-anonymity, single valued attributes are replaces by a range of values in which that single value would occur, for e.g. if Age had value 33, it is now replaced with Age falling in [30-39].

5.2. ℓ-diversity

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

ℓ-diversity provides a higher level of protection as compared to *k*-anonymity. Although *k*-anonymity is adequate for mobile applications that provide protection against disclosure of the mobile user identity, however, if the attacker has sufficient background knowledge about the user, they can infer sensitive information about the user, irrespective of the fact that their data has been *k*-anonymized by quasi-identifiers. This situation is known as a background knowledge attack. This would also occur when the sensitive attribute value is the same for the group of quasi-identifiers that the user belongs to. This is known as a homogeneity attack.

The following example illustrates the difference between these two attacks. Table 4 is the original patients table containing information about the patient's zip code, age and the sensitive

Table 4. 3 Anonymized Version of Patient Table

	ZIP Code	Age	Disease
1	476**	2*	Heart Disease
2	476**	2*	Heart Disease
3	476**	2*	Heart Disease
4	4790*	>=40	Flu
5	4790*	>=40	Heart Disease
6	4790*	>=40	Cancer
7	476**	3*	Heart Disease
8	476**	3*	Cancer
9	476**	3*	Cancer

attribute being diagnosed disease. After a 3-anonymity transformation on this table we get 3 equivalence classes of quasi-identifiers based on zip codes & age.

Looking at the first equivalence class, it is evident that the sensitive attribute has the same value for all members of that equivalence class. This is an example of a homogeneity attack since given that person A is 22 years old and belongs to zip code 47602, it can be successfully concluded that A is diagnosed with Heart Disease.

Now, looking at the last equivalence class, the sensitive attribute is not the same for all the records. If the attacker has some background knowledge on the sensitive attribute they can infer the required information about the user. Given that person B is 32 years old and lives in zip code 47607, besides this, the attacker also knows that person B has a low risk of Heart Disease, then the attacker can successfully conclude that person B must have Cancer.

Table 5. Original Patient Table

	ZIP Code	Age	Disease
1	47677	29	Heart Disease
2	47602	22	Heart Disease
3	47678	27	Heart Disease
4	47905	43	Flu
5	47909	52	Heart Disease
6	47906	47	Cancer
7	47605	30	Heart Disease
8	47673	36	Cancer
9	47607	32	Cancer

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

These two attacks show the limitations of *k*-anonymity when it comes to privacy defense mechanisms for attribute disclosure. Several authors, e.g. [15][16][17], have addressed these limitations. In [15], Machanavajjhala et. al provide ℓ-diversity as a solution to these two problems with *k*-anonymity. The ℓ-diversity solution is represented as a general principle. This principle requires that the equivalence class represented by a given quasi-identifier group, have at least ℓ number of diverse values for its sensitive attributes. If this is true for every equivalence class of the table then that table is said to have ℓ-diversity. Four types of ℓ-diversity solutions are discussed in [15] in which the author defines the sensitive attribute to be "well-represented".

5.2.1. Distinct ℓ-diversity

Here, at least ℓ numbers of distinct values are set for the sensitive attribute in each and every equivalence class. This means that the number of values a sensitive attribute can take can be fixed, however the frequency with which they occur vary and can still cause an attack. Therefore, Distinct ℓ-diversity is not helpful in preventing probabilistic inference attacks. An attacker can observe that a particular value for the sensitive attribute has a higher probability of occurring in a specific equivalence class and victim is therefore probable to have that same value as well. Stronger versions of ℓ-diversity have thus been developed to overcome this as given below.

5.2.2. Probabilistic ℓ-diversity

To deal with the problem described above, a restriction of ℓ=1 is put on the number of times a distinct value of a sensitive attribute is to occur in an anonymized table. This means the sensitive value can occur more than $1/\ell$ times. The attacker is therefore not able to derive the sensitive value of an individual with probability greater than $1/\ell$.

5.2.3. Entropy ℓ-diversity

$$Entropy(E) = -\sum_{s \epsilon S} p(E,s) \log p(E,s)$$

This kind of ℓ-diversity is calculated using the formula given below, where,
E: Equivalence Class
s: Sensitive Attribute Value
S: Domain of the sensitive attribute
p(E,s): Fraction of those records within E having value s.
If for every equivalence class E, Entropy (E)>=log ℓ then that table is said to have entropy ℓ-diversity. This is sometimes considered to be too restrictive, as the entropy of the entire table may be low if a few values are very common.

5.2.4 Recursive (c,ℓ)-diversity

A less restrictive version of ℓ-diversity is Recursive (c,ℓ)-diversity. The following condition is to hold true in order for a table to have this type of diversity in its equivalence classes,

$r_1 < c(r_\ell + r_{\ell+1} + ... + r_m)$, and

m: number of values in an equivalence class E

r_i: frequency of the i^{th} most frequent sensitive value within an equivalence class E, where $1 <= i <= m$,

Recursive (c,ℓ)-diversity (c is a float number and ℓ is an integer) makes sure that the most frequent value does not appear too frequently, and the less frequent values do not appear too rarely. A table is said to have recursive (c,ℓ)-diversity if all of its equivalence classes have recursive (c,ℓ)-diversity.

5.3. *t*-Closeness

Here a threshold *t* is defined, that cannot be exceeded if *t*-closeness is to be achieved. *t* is a measure of the distance between the distribution of a sensitive attribute within an equivalence class and the distribution of the attribute in the whole table. A table is said to have *t*-closeness if all equivalence

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

classes have *t*-closeness. [32] discuss solutions to limitations of this approach to data privacy as well and suggest (*n*,*t*) closeness as an enhanced privacy technique for mobility data.

5.4. Mix Zones

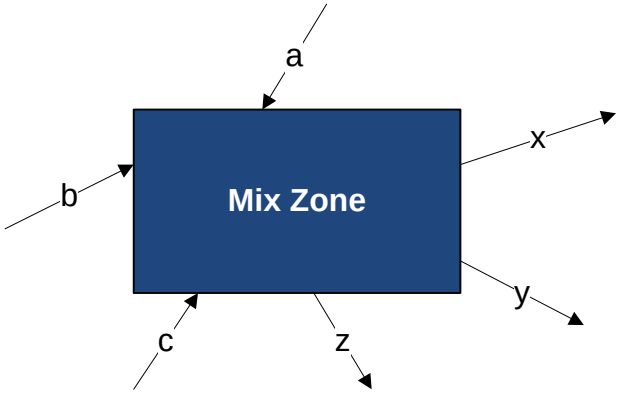

Figure 5. Mix Zones

Another way to protect users is to provide them anonymity over a specific area or zone called mix zone. This zone is responsible for randomizing the incoming and outgoing messages using normal message routers as well as mix routers. The basic idea is that a mix-router collects *k* equal length packets as input and reorders them randomly before forwarding them, thus ensuring *unlink-ability* between incoming and outgoing messages. This concept has been extended to LBS, namely, mix-zones [18].

For example, in the figure above, the shaded region is the mix zone. The outgoing users x, y, z could be either one of the incoming users a, b c. The users are unlink-able with their previous identities. When users a, b, c enter the mix-zone, they change to a new, unused pseudonym. Also, no location information is sent to any location-based application when they are in the mix-zone. When an attacker sees a user *z* exit from the mix-zone, they are unable to distinguish *z* from any other user who was in the mix-zone with *z* at the same time. A set of users S is said to be *k*-anonymized in a mix-zone Z if all following conditions are met [19]:

1. The user set S contains at least *k* users, i.e., $|S| \geq k$.
2. All users in S are in Z at a point in time, i.e., all users in S must enter Z before any user in S exits.
3. Each user in S spends a completely random duration of time inside Z.
4. The probability of every user in S entering through an entry point is equally likely to exit in any of the exit points.

Table 6 gives an example of 3-anonymity for the mix-zone depicted in the mix zone figure, where three users with real identities, α, β, and γ enter the mix-zone with old pseudonyms (P_{old}) a, c, and b at

Table 6. 3-Anonymity for MixZone in Figure 5

User ID	P_{old}	P_{new}	ts_{enter}	ts_{exit}	t_{inside}
α	a	y	2	9	7
β	c	x	5	8	3
γ	b	z	1	11	10

timestamps (ts_{enter}) 2, 5, and 1, respectively. Users α, β, and γ exit the mix-zone with new pseudonyms (P_{new}) y, x, and z at timestamps (ts_{exit}) 9, 8, and 11, respectively. Thus, they all are in the mix-zone during the time period from 5 to 8. Since they stay inside the mix-zone with random time periods (i.e., t_{inside}), there is a strong unlink-ability between their entry order ($\gamma \rightarrow \alpha \rightarrow \beta$) and exit order ($\beta \rightarrow \alpha \rightarrow \gamma$) [4].

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

5.5. Spatial Cloaking

In mix-zones, a specified spatial region was defined, within which the users' identities were replaced. In spatial cloaking, instead of replacing the identifiers of the users, the location of the users is replaced with the location of the broader region that the user is in. This broader region is given the term 'cloaking region' (CR). And this process of transformation is known as Spatial Cloaking. It is one of the most popular and intuitive forms of data transformations. Having the GPS of their wireless device activated, the user walking through a shopping district in the city centre, might have their location given to an LBS replaced from the exact location or name of the shopping centre to a more broader name like 'CityCentre'. This region could not only be one that covers the nearby building blocks and streets but may be a predefined region, e.g., "Citycenter". The cloaking region CR (p) of a point p is created in such a way that it validates a certain privacy predicate PP, i.e., PP (CR) = true. For example, PP might require that the number of users who exist in CR is over *k*, thus guaranteeing *k* anonymity [3].

Spatial cloaking takes the generalization concept of k-anonymity and applies it to locations instead of ranges of quasi-identifiers. It is an adjusted form of the generalization technique used for relational [23], transactional [24] and other types of data. In the previous examples we saw quasi-identifiers of numeric data be changed to numeric ranges, e.g. 33 to [33→35] or categorical data according to some predefined hierarchy, e.g. "blood cancer" to "cancer". In order to replace cloaking regions, the transformation takes place on the basis of the privacy requirements of the data publisher or to predefined regions with known characteristics. This involves a mapping of the exact location of users, expressed usually by two coordinates (or three if they are time stamped), onto a larger set of co-ordinates of which the original may be a subset. One approach in creating the CR is to use a predefined grid for the map where the users move. [3] The user is assigned the grid cell or many cells that satisfy the data publisher's PP. Several approaches use a hierarchical organization of grids cells [28; 22]. Partitioning the map to predefined cells is a popular method since it is computationally less expensive than creating arbitrary regions. The second approach is to create arbitrary regions that can provide better utility to the anonymized data, since the algorithm can enlarge CR only as much as needed to validate PP and it is not constrained by the grid granularity. [3] The downside is that it is computationally more expensive and it is more prone to minimality attacks [27] as shown in [25; 26]. It is adopted by numerous anonymization methods that address both identity and location disclosure.

Note that the CR might be a set of disjoint regions and it might even not contain the original location [29]. This technique has variations that are applied to both snapshot LBS as well as continuous LBS. In the case of continuous LBS, 3 techniques are discussed in [4] to cloak trajectories.

5.6. Path Confusion

This technique aims to protect users from adversaries with knowledge of a moving target. The user trajectories are aimed at by attackers using target tracking algorithms. Here the background knowledge of the adversary consists of the speed of the moving target and direction to be fed into the tracking algorithms to predict with a certain level of accuracy the likely location of the target at a future timestamp. An easy landscape for such an attack is usually a road network where the speeds and directions of a vehicle are constrained by the underlying traffic network. These algorithms undermine the use of anonymization techniques since consecutive location samples from a vehicle are temporally and spatially correlated, trajectories of individual vehicles can be constructed from a set of location samples with anonymized pseudonyms reported from several vehicles through the target tracking algorithms [34]. These location samples (or the one with the highest probability if there are multiple candidate location samples) are used to link to the same vehicle through Maximum Likelihood Detection [34].

Path Confusion aims at avoiding this link-ability of consecutive location samples to individual vehicles with high certainty [35]. The technique employs a time-to-confusion and a tracking uncertainty to determine a confusion level for a user trajectory. The time-to-confusion is considered the degree of privacy of the path confusion. It is the tracking time between two location samples where an attacker is

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

unable to determine the next sample with sufficient tracking certainty. Tracking uncertainty is computed by $H = -\Sigma\ p_i \log p_i$, where p_i is the probability that location sample *i* belongs to a target vehicle. Smaller values of H mean higher certainty or lower privacy. Given a maximum allowable time to confusion, *ConfusionTime*, and an associated uncertainty threshold, *ConfusionLevel*, a vehicle's location sample can be safely revealed if the time between the current time *t* and the last point of its confusion is less than *ConfusionTime* and tracking uncertainty of its sample with all location samples revealed at time *t* is higher than *ConfusionLevel*. To reduce computational overhead, the computation of tracking uncertainty can only consider the *k*-nearest location samples to a predicted location point (calculated by the target tracking algorithm), rather than all location samples reported at time *t*. [4]

5.7. Dummy Trajectories

As the name suggests, this technique protects user trajectories by letting a mobile user generate fake location trajectories, called dummies, to protect trajectory privacy [33]. [4] describes the process with the example given below. Given a real user location trajectory T_r and a set of user-generated dummies T_d, the degree of privacy protection for the real trajectory is measured by the following metrics [33]:

1. Snapshot disclosure (SD). Let m be the number of location samples in T_r, S_i be the set of location samples in T_r and any T_d at time t_i, and $|S_i|$ be the size of S_i. SD is defined as the average probability of successfully inferring each true location sample in T_r, i.e., SD = $\frac{1}{m}\sum_{i=1}^{m}\frac{1}{|S_i|}$.

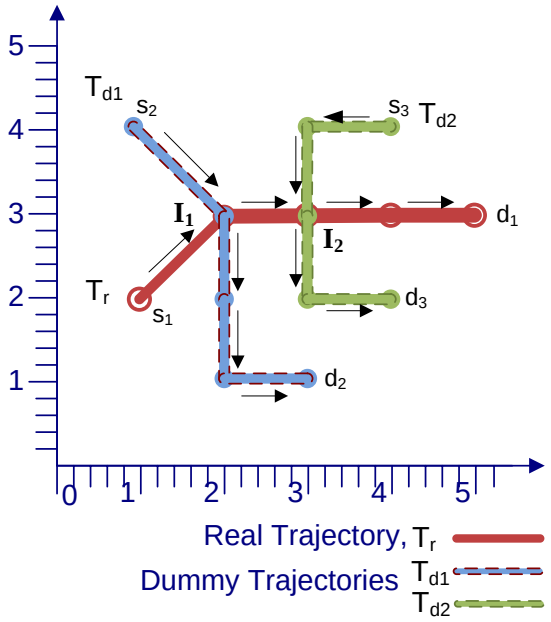

Figure 6. One Real Trajectory T_r & Two Dummy Trajectories, T_{d1} & T_{d2}

Figure 6 gives a running example of n = 3 trajectories and m = 5 location samples, where given the starting points s_1 (1,1), s_2 (1,4) & s_3 (4,4) and ending points d_1 (5,3), d_2 (4,2) & d_3 (3,1); T_r is from location s_1 to location d_1 (i.e., $s_1 \rightarrow d_1$), T_{d1} is $s_2 \rightarrow d_2$, and T_{d2} is $s_3 \rightarrow d_3$. There are two intersections I_1 and I_2. At time *i* = 1, since there are three different locations, i.e., (1, 2), (1, 4) and (4, 4), $|S_1| = 3$. Thus,

$$SD = \frac{1}{5}\left(\frac{1}{3}+\frac{1}{2}+\frac{1}{2}+\frac{1}{3}+\frac{1}{3}\right)=\frac{2}{5}.$$

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

2. Trajectory disclosure (TD). Given n trajectories, where k trajectories have intersection with at least one other trajectory and $n-k$ trajectories do not intersect any other trajectory, let N_k be the number of possible trajectories among the k trajectories. TD is defined as the probability of successfully identifying the true trajectory among all possible trajectories is $\frac{1}{N_k+(n-k)}$.
In the running example, $N_k = 3$ and there are eight possible trajectories, i.e.,
$s_1 \rightarrow I_1 \rightarrow d_2$,
$s_1 \rightarrow I_1 \rightarrow I_2 \rightarrow d_1$,
$s_1 \rightarrow I_1 \rightarrow I_2 \rightarrow d_3$,
$s_2 \rightarrow I_1 \rightarrow d_2$,
$s_2 \rightarrow I_1 \rightarrow I_2 \rightarrow d_1$,
$s_2 \rightarrow I_1 \rightarrow I_2 \rightarrow d_3$,
$s_3 \rightarrow I_2 \rightarrow d_1$,
and $s_3 \rightarrow I_2 \rightarrow d_3$. Hence, TD = $\frac{1}{8+(3-3)} = \frac{1}{8}$.
3. Distance deviation (DD). DD is defined as the average distance between the i^{th} location samples of T_r and each T_{di}, i.e.,

$$DD = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{n}\sum_{j=1}^{n} dist\left(T_r^i, T_{d_j}^i\right)\right)$$

, where dist(p, q) denotes the Euclidean distance between two point locations p and q. In the running example,

DD = $\frac{1}{5}$ × (2.80 + 0.71 + 0.71 + 2.12 + 2.12) = 1.69.
Given a real trajectory T_r and the three user-specified parameters SD, TD, and DD in a privacy profile, the dummy-based anonymization algorithm incrementally uses DD to find a set of candidate dummies and selects one with the best matching to SD and TD until it finds a set of trajectories (including T_r and selected dummies) that satisfies all the parameters [33].

6. Conclusion and Future Work

The paper attempts to cover the research in the two principal directions of privacy for mobility data: user privacy in location based services, and anonymity in the publication of personal mobility data. Furthermore, it considers privacy issues in emerging applications, such as location based social networks. Applications arising from mobility data pose unique challenges to the data mining community. Data mining techniques, when applied on data collected by location based applications, have a lot of potential in supporting decision making in tasks such as urban planning, intelligent transportation, and environmental pollution. However, privacy-enhancing methods are necessary to ensure that the collected data are protected against privacy threats.

6.1 Limitations & Future Work

This research's contribution has helped in highlighting a set of algorithms that can be proven beneficial to privacy enhancement of mobility users. However, given the scope of this research puts limitations on the analysis of each technique. The researchers were limited in their access to a mobile dataset to implement the algorithms and compare and contrast the performance of each. Given access to such mobility dataset would then further enhance research in this direction by comparing implementations of the algorithms brought forward in this paper and can be implemented as future work.

In future work, we also suggest a comparative analysis that maybe conducted on the suggested algorithms and a hybrid algorithm be proposed, catering to user data and trajectory data which can be experimented for performance or overhead based on the findings. In this way, this research paper

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

provides a sound review base to help drive future work in privacy algorithms for mobility data and their mining.

Other open research areas identified are found by addressing the limitations of each technique, where the selection of appropriate quasi-identifiers could be further explored as a research area for *k*-anonymity and ℓ-diversity.

Another open research area within ℓ-diversity is the exploration of parameters to define optimum frequency of sensitive values in k-anonymized tables to prevent homogeneity attacks is another research area. Specifically, Recursive ℓ-diversity can be considered, where appropriate definitions for too rare and too frequent sensitive values may be proposed, giving rise to such an optimum frequency parameter.

References

[1] Usama Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, "From Data Mining to Knowledge Discovery in Databases", Journal of Association of Advancement of Artificial Intelligence (AI Magazine), vol.17, no.3, pp.37-54, 1996

[2] Aris Gkoulalas-Divanis, Yucel Saygin, Dino Pedreschi , "Privacy in Mobility Data Mining", In ACM SIGKDD Explorations Newsletter, (ACM), vol.13, no.1, pp.4-5, 2011

[3] Manolis Terrovitis, "Privacy preservation in dissemination of location data", In ACM SIGKDD Explorations Newsletter, (ACM), vol.13, no.1, pp.6-18, 2011

[4] Chi-Yin Chow, Mohamed F. Mokbel , "Trajectory privacy in location based services and data publications", In ACM SIGKDD Explorations Newsletter, (ACM), vol.13, no.1, pp.19-29, 2011

[5] Latanya Sweeney, "*k*:Anonymity: A Model for Protecting Privacy", International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems [IJUFKBS], (World Scientific), vol. 10, no.5, pp. 96, 2002

[6] B. Chor , O. Goldreich , E. Kushilevitz , M. Sudan, "Private information retrieval", In Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS'95), pp.41, 1995

[7] M. L. Damiani, E. Bertino, and C. Silvestri, " The PROBE framework for the personalized cloaking of private locations", Transactions on Data Privacy , Issue 3, vol. 2, pp. 123–148, 2010

[8] Gabriel Ghinita , Maria Luisa Damiani , Claudio Silvestri , Elisa Bertino, "Preventing velocity-based linkage attacks in location-aware applications", In Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 246–255, 2009

[9] E. Kushilevitz , R. Ostrovsky, "Replication is not needed: single database, computationally-private information retrieval", In Proceedings of the 38th Annual Symposium on Foundations of Computer Science (FOCS '97), pp.364, 1997

[10] S. Mascetti, C. Bettini, D. Freni, X. S. Wang, S. Jajodia, "Privacy-Aware Proximity Based Services," In Proceedings of 10th International Conference on Mobile Data Management: Systems, Services and Middleware (MDM), pp.31-40, 2009

[11] L. Siksnys, J. R. Thomsen, S. Saltenis, and M. L. Yiu ,"Private and Flexible Proximity Detection in Mobile Social Networks", In Proceedings of 11th International Conference on Mobile Data Management [MDM], (IEEE Computer Society Press), pp. 75-84, 2010

[12] M. L. Yiu, C. S. Jensen, X. Huang, and H. Lu, "Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services," In Proceedings of International Conference on Data Engineering [ICDE], (IEEE Computer Society), pp. 366–375, 2008

[13] A. Narayanan, N. Thiagarajan, M. Lakhani, M. Hamburg, and D. Boneh, "Location privacy via private proximity testing", In 18th Proceedings of Annual Network & Distributed System Security Symposium [NDSS], 2011

[14] G. Ghinita, P. Kalnis, and S. Skiadopoulos, " PRIVE: Anonymous location-based queries in distributed mobile systems" , In Proceedings of the 16th International Conference on World Wide Web, pp. 371 – 380, 2007

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: 10.7321/jscse.v3.n1.2

[15] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "ℓ-diversity: Privacy Beyond *k*-Anonymity", ACM Transactions on Knowledge Discovery from Data [TKDD], vol.1 no.1, p.3-es, 2007
[16] T.M. Truta, B. Vinay, "Privacy Protection: P-Sensitive *k*-Anonymity Property," In Proceedings of 22nd International Workshop on Privacy Data Management (ICDE Workshops), pp.94, 2006
[17] X. Xiao,Y. Tao, "Personalized Privacy Preservation," In Proceedings of ACM SIGMOD, pp. 229-240, 2006
[18] A. R. Beresford, F. Stajan, "Location privacy in pervasive computing", Pervasive Computing, (IEEE), vol.2, no.1, pp. 46- 55, 2003
[19] B. Palanisamy, L. Liu. Mobimix, "Protecting location privacy with mix zones over road networks", In Proceedings of the IEEE 27th International Conference on Data Engineering [ICDE], pp.494-505, 2011
[20] Francesco Bonchi, Laks V. S. Lakshmanan and Hui (Wendy) Wang, "Trajectory anonymity in publishing personal mobility data", In ACM SIGKDD Explorations Newsletter, vol.13, no.1, pp.30-42, 2011
[21] Leyla Kazemi, Cyrus Shahabi, "A privacy-aware framework for participatory sensing", In ACM SIGKDD Explorations Newsletter, vol.13, no.1, pp.43-51, 2011
[22] M. Gruteser, D. Grunwald, " Anonymous usage of location-based services through spatial and temporal cloaking", In Proceedings of the 1st international conference on Mobile systems, applications and services [MobiSys '03] , pp. 31–42, 2003
[23] K. LeFevre, D. J. DeWitt, R. Ramakrishnan, "Incognito: Efficient full-domain k-anonymity", In Proceedings of ACM International Conference on Management of Data (SIGMOD), pp. 49–60, 2005
[24] M. Terrovitis, N. Mamoulis, P. Kalnis, "Privacy-preserving Anonymization of Set-valued Data", VLDB , vol.1, no.1, 2008
[25] Gabriel Ghinita, Keliang Zhao, Dimitris Papadias, Panos Kalnis, "A reciprocal framework for spatial K-anonymity", Journal of Information Systems, vol.35, no.3, pp.299-314, 2010
[26] Panos Kalnis , Gabriel Ghinita , Kyriakos Mouratidis , Dimitris Papadias, "Preventing Location-Based Identity Inference in Anonymous Spatial Queries", IEEE Transactions on Knowledge and Data Engineering, vol.19, no.12, pp.1719-1733, 2007
[27] R. C. W. Wong, A. W.C. Fu, K. Wang, and J. Pei, " Minimality attack in privacy preserving data publishing", In Proceedings of VLDB Endowment , pp. 543–554, 2007
[28] M. F. Mokbel, C.Y. Chow, W. G. Aref, "The new casper: A privacy-aware location-based database server", In Proceedings of International Conference on Data Engineering [ICDE], pp. 1499-1500, 2007
[29] M. Duckham , L. Kulik, "A formal model of obfuscation and negotiation for location privacy", In Proceedings of 3rd International Conference on Pervasive Computing (Pervasive), pp. 152–170, 2005.
[30] C.Y. Chow, M. F. Mokbel, "Enabling private continuous queries for revealed user locations", In Proceedings of the International Symposium on Spatial and Temporal Databases [SSTD], pp. 258-275, 2007
[31] T. Xu, Y. Cai, "Location anonymity in continuous location-based services", In Proceedings of the ACM Symposium on Advances in Geographic Information Systems, Article 39, pp.1-8, 2007.
[32] Ninghui Li, Tiancheng Li, Suresh Venkatasubramanian, "Closeness: A New Privacy Measure for Data Publishing", IEEE Transactions on Knowledge and Data Engineering [TKDE], Vol. 22, No. 7, pp.943-956, 2010
[33] T. H.You, W. C. Peng, W. C. Lee, " Protecting moving trajectories with dummies", In Proceedings of the International Workshop on Privacy-Aware Location-Based Mobile Services, pp.278-282, 2007.
[34]M. Gruteser, B. Hoh, "On the anonymity of periodic location samples", In Proceedings of the International Conference on Security in Pervasive Computing, pp.179-192, 2005.

International Journal of
Soft Computing and Software Engineering (JSCSE)
e-ISSN: 2251-7545
Vol.3,No.1, 2013
Published online: Jan 25, 2013
DOI: **10.7321/jscse.v3.n1.2**

[35] B. Hoh, M. Gruteser, H. Xiong, A. Alrabady, "Achieving guaranteed anonymity in GPS traces via uncertainty-aware path cloaking", IEEE Transactions on Mobile Computing, vol. 9, no.8, pp.1089–1107, 2010.
[36] Mehdi Alimi Motlagh Fard, Hamid Reza Ranjbar, Abbas Davani, Mehdi Sadegh Zadeh, "A Data Mining Approach To Gen Dynamic Behavioral Process", International Journal of Soft Computing and Software Engineering [JSCSE], vol.1, no.1, pp.18-24, 2011
[37] Ladan Darougaran, Hossein Shahinzadeh, Mohammadreza Salavati, "Simulated Annealing algorithm for Data Aggregation Trees in Wireless Sensor Networks", International Journal of Soft Computing and Software Engineering [JSCSE], vol.1, no.1, pp.36-43, 2011