A Basic Framework for Privacy Protection in Personalized Information Retrieval:

An Effective Framework for User Privacy Protection

Zongda Wu, Shaoxing University, China

Shigen Shen, Shaoxing University, China

https://orcid.org/0000-0002-7558-5379

Huxiong Li, Shaoxing University, China

Haiping Zhou, Shaoxing University, China

Chenglang Lu, Zhejiang Institute of Mechanical and Electrical Engineering, China*

ABSTRACT

Personalized information retrieval is an effective tool to solve the problem of information overload. Along with the rapid development of emerging network technologies such as cloud computing, however, network servers are becoming more and more untrusted, resulting in a serious threat to user privacy of personalized information retrieval. In this paper, the authors propose a basic framework for the comprehensive protection of all kinds of user privacy in personalized information retrieval. Its basic idea is to construct and submit a group of well-designed dummy requests together with each user request to the server to mix up the user requests and then cover up the user privacy behind the requests. Also, the framework includes a privacy model and its implementation algorithm. Finally, theoretical analysis and experimental evaluation demonstrate that the framework can comprehensively improve the security of all kinds of user privacy, without compromising the availability of personalized information retrieval.

KEYWORDS

Algorithm, Constraint, Framework, Information Retrieval, Personalized, Privacy Model, User Privacy

INTRODUCTION

Along with the rapid development of network technology, the amount of information on the Internet is expanding rapidly, which leads to the serious problem of information overload (Wu et al., 2021a; Lin et al, 2021), and in turn the bottleneck of end users' effective use of information resources on the Internet (Wu et al., 2021b, 2021c, 2021d). Based on the specific information needs of users (such as points of interest, locations and preferences), personalized information retrieval can provide users with resources to meet their personalized needs, and then help users quickly obtain the target data from massive resources, thus it is an effective tool to solve the problem of information overload (Zhou et al., 2020; Zhang et al., 2020), and has attracted wide attention from both scientific and industrial

DOI: 10.4018/JOEUC.292526

*Corresponding Author

communities. However, on the one hand, along with the rapid development of emerging network technologies such as cloud computing, the server of personalized information retrieval is becoming more and more untrusted, and has become the main source of end user privacy disclosure (Liu et al., 2018; Such & Natalia, 2018). On the other hand, in order to obtain accurate service results, personalized information retrieval requests issued by users to the server certainly would contain a large number of sensitive information (such as interests, preferences and locations). All of this information would be collected by the untrusted server, which is bound to pose a serious threat to the user privacy (Wu et al., 2021e; Liu et al., 2021; Saura et al., 2021). Therefore, the problem of privacy and security has become a major obstacle to the further development and application of personalized information retrieval services on the Internet (Wang et al., 2019; Hewitt & White 2021), and thus has become an important topic in the field of organizational and end user computing.

In this context, this paper focuses on the privacy protection of personalized information retrieval, and proposes an effective solution. To obtain personalized results, it is required for each user to not only report his current geographic location (by using query locations) and personal preference (by using preference profiles) to the untrusted server, but also report the content that he wants to obtain (by using query points of interest). Therefore, the user privacy that needs to be protected in personalized information retrieval mainly includes location privacy (which can be obtained by analyzing query locations), query privacy (which can be obtained by analyzing query interest points) and preference privacy (which can be obtained by analyzing preference profiles). To this end, this paper focuses on the comprehensive protection of all kinds of user privacy in personalized information retrieval, whose goal can be summarized as follows. According to the distribution features of all kinds of request data (including preferences, locations and interest points) related to personalized information retrieval, we aim to construct a basic framework for user privacy protection, so as to overcome the application limitations of existing technical methods in personalized information retrieval, i.e., to comprehensively improve the security of users' preference privacy, location privacy and query privacy on the untrusted server, without compromising the availability of an existing personalized information retrieval platform.

The main contributions of this paper are threefold. (1) A unified framework for the user privacy protection of personalized information retrieval, which has good practical usability. (2) A privacy model for the user privacy protection of personalized information retrieval, which formulates the constraints that should be satisfied for the effective protection of preference privacy, location privacy and query privacy. (3) An implementation algorithm for the privacy model under the framework, and can comprehensively improve the security of all kinds of user privacy on the untrusted server. Overall, this paper presents an important and valuable study attempt to the protection of user privacy in personalized information retrieval, and its study result is of a positive influence on the problem of privacy and security in the field of organizational and end user computing.

LITERATURE REVIEW

The right of privacy is one of citizens' personal rights, so it is a basic social consensus to its protection. Along with the continuous advancement of informatization construction, the problem of user privacy is becoming more and more prominent. Due to strong administrative binding force, laws and regulations are regarded as one of powerful tools to protect civil rights (Wu et al., 2021d). Therefore, scholars from the field of social sciences are more concerned with the problem of user privacy from the perspective of legislation (Wu et al., 2019a, 2019b, 2021d). At present, governments in the world have issued laws and regulations related to the right of citizens' privacy at multiple levels. However, making laws and regulations cannot solve the problem of user privacy fundamentally. The main reason is that the effectiveness of laws and regulations is based on a precondition that they can be obeyed strictly by service providers and their administrators (based on an assumption that each server is highly trusted). However, a large number of privacy leakage incidents show that the possibility of administrators snooping users' privacy information is widespread, driven by interests (Wu et al., 2018a; Li et al., 2019; Dong et al., 2020). Now, along with the rapid development of emerging network technologies such as cloud computing, a large number of servers are deployed to the cloud, which further reduces the credibility of servers, and increases the disclosure risk of user privacy. Therefore, the protection of user privacy in personalized information retrieval needs to be solved by not only legislations, but also advanced technical means. Although traditional security means (such as identity authentication (Shen et al., 2019) and access control (Shen et al., 2018) can prevent the access to unauthorized data by external users, their effectiveness is also based on the assumption that each server is highly trusted (Wu et al., 2018a), i.e., they have the same limitations as legislations.

However, for the problem of user privacy protection in an untrusted network environment, scholars in the field of information sciences have presented a number of effective solutions, including encryption methods, pseudonym methods, obfuscation methods and confusion methods. In this section, we review these methods, and then analyze their application limitations in personalized information retrieval.

Encryption Methods

Here, encryption refers to encrypting the privacy data contained in user service requests to make it invisible to the server, and achieve the purpose of privacy protection (Baumeler & Brodbent, 2014). Encryption methods can be further divided into privacy information retrieval and cryptographic protocol. The privacy information retrieval protocol (Mei et al., 2018) was first of all used to access user data in an outsourced network environment, which allows users to fetch their desired data from a database on the premise that the server cannot know their particular requests. However, it is difficult for privacy information retrieval to be applied to modern information retrieval, due to its high complexity and its inability of the server to carry out targeted advertising. The cryptographic protocol is mainly designed for textual retrieval (Zhang et al., 2016). After some extension, this kind of method can help users to retrieve the target text documents that fully meet a given keyword Boolean expression. However, it is difficult for the cryptographic protocol to be applied to modern text retrieval, since modern text retrieval needs to retrieve the most similar text document with a user query, i.e., modern text retrieval needs to support similarity text retrieval, rather than deterministic retrieval (Wu et al., 2015). In summary, this kind of method not only requires the assistance of high complexity algorithms, but also requires some change to information service algorithms, leading to the change of the whole architecture of an information service platform, and in turn reducing its actual usability in personalized information retrieval.

Pseudonym Methods

For a pseudonym method, its basic idea is to replace the user identity in a service request with a temporary pseudonym to break the natural connection between a user and his service request (Ravi et al., 2019; Wu et al., 2019a; Shuai et al., 2021). It is easy for this kind of method to integrate into an existing information service to protect users' privacy. However, this kind of method is heavily dependent on the effectiveness of pseudonyms (Liu et al., 2018a). Therefore, to enhance the effectiveness of pseudonyms, a mixed region was proposed (Ziegeldorf et al., 2017; Li et al., 2017; Arain et al., 2017), which refers to a specific region where multiple users change their pseudonyms in a centralized way, and these users cannot submit service requests or receive service information, making it difficult for an attacker to track the users. However, users in a mixed region cannot communicate with each other, which inevitably would reduce the quality of information service. However, using pseudonyms alone cannot protect user privacy fully, since it makes no change to user request data (such as location and query text), which makes an attacker still have some probability to determine the user real identity according to the user requests themselves. Also, it is an obstacle for a pseudonym method to an application scenario that requires user identity authentication. In most of existing personalized information retrieval platforms, it is required for users to login with real names, so a pseudonym method is not suitable for the protection of user privacy in personalized information retrieval.

Obfuscation Methods

An obfuscation method is mainly designed for the protection of location privacy, whose basic idea to generalize or perturb query location information, to make it difficult for an attacker to identify the exact location of a particular user (Ruchika & Rao, 2017; Wang et al, 2018). Here, generalization refers to replacing each user real location with a generalized region (called a cloaking region), which is generally constructed by a trusted anonymous third-party server through using the k-anonymity (Zeng et al., 2018; Soma et al., 2017; Xue et al., 2017). However, traditional methods for the generation of k-anonymous cloaking regions (Soma et al., 2017; Xue et al., 2017) are difficult to achieve a predetermined level of privacy protection in continuous space queries. Perturbation refers to adding some errors or noises (Dewri, & Thurimella 2016) into each user query in a controllable way. In order to provide stricter privacy protection, recent studies attempt to use the differential privacy model to control the amount of noises added into continuous queries, where the representative ones are the spatial indistinguishability model (Chatzikokolakis et al., 2015) and its derivative model (Wang et al., 2018). In summary, this kind of method can meet the constraints of efficiency and accuracy of personalized information retrieval, but in general, it is targeted only for one single type of user privacy (location privacy), i.e., it cannot improve the security of all kinds of user privacy comprehensively in personalized information retrieval.

Confusion Methods

A confusion method refers to confusing each user service request before exposing it to the server, so as to make it difficult for the untrusted server to know the user real request, thus achieve the protection to user privacy. Aiming at text retrieval services, TrackMeNot (Meng et al., 2016) proposes to hide each user query in randomly constructed dummy queries, so as to make it difficult for the server to find out the user queries. In (Pang et al., 2012), an evaluation model for the protection of user topic privacy was further proposed, which allows users to specify the protection degree of query topics. However, because this kind of method does not takes into consideration the distribution features of user data, the constructed dummy queries can easily be ruled out, thus failed to confuse and protect user privacy. To solve this, Arampatzis et al., 2016 proposed to construct a group of static queries in advance, and then Wu et al., 2020a proposed to construct a generalized dummy query to further improve the confusion effect of dummy queries to user privacy. However, the main problem of this kind of method is that using a dummy to replace a user query would reduce the accuracy. Also, some scholars try to adopt a confusion method to protect users' preference privacy (Wu et al., 2018b, 2020c) and location privacy (Niu et al., 2014). In summary, a confusion method is developed based on a client-based architecture, independent of a third-party server, so it has good practical usability. However, most of existing methods do not take into full consideration the feature distribution of user requests, which greatly reduces the construction quality of dummy requests. In addition, most of existing methods are only targeted for a single kind of user privacy, so they cannot meet the privacy requirements of personalized information retrieval.

Problem Analysis

From the above, we know that although there have been a number of effective technical methods proposed for the protection of user privacy under an untrusted network environment, generally, they are designed only for a single kind of user privacy, consequently, making them difficult to meet the actual needs of personalized information retrieval in terms of usability, accuracy, efficiency and security. Aiming at the protection of user privacy in personalized information retrieval, there are still many problems to be solved.

P1: We should pay attention not only to the protection of user privacy, but also to the usability, accuracy and efficiency of personalized information retrieval. In general, personalized information retrieval exists as an important part of a large-scale network platform. However, to protect user privacy, most of existing methods require some changes to the existing platform or algorithm, or require some compromises to the accuracy and efficiency of information retrieval, thereby, reducing the practical usability of the methods. Therefore, it is an important problem that the protection of user privacy should be built on no compromise to the usability, accuracy and efficiency of personalized information retrieval.

P2: We should pay attention not only to the privacy protection of users' current requests, but also to the privacy protection of users' historical requests. In personalized information retrieval, a sequence of users' service requests exhibit regular distribution features (non-random). For example, a user usually likes to search around fixed topics in a certain period of time. However, most of existing methods are only targeted for the protection of users' current requests, without taking into consideration users' historical requests, greatly reducing the effect of privacy protection. Therefore, the protection of user privacy should be built on a sequence of user requests, and should take into full consideration the distribution features of a user request sequence, so as to improve the effect of privacy protection.

P3: We should not only pay attention to a single type of user privacy in personalized information retrieval, but also construct a comprehensive model to protect all types of user privacy as a whole. Most of existing methods are **targeted** only for one single type of user privacy (e.g., only for location privacy). However, in personalized information retrieval, service request data issued by users contain various user privacy (such as preference privacy, location privacy and query privacy), and there exist strong semantic correlations between them. All of these require that we should not consider a single type of user privacy in isolation. Instead, we should fully take into account the semantic correlations between different kinds of user privacy, to comprehensively improve the security of all kinds of user privacy of personalized information retrieval.

PRELIMINARY

System Model

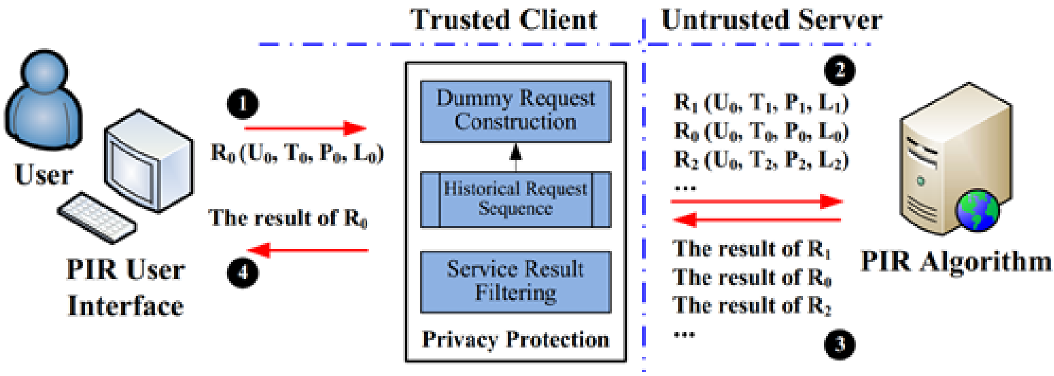
Personalized information retrieval, according to the points of interest, preference profile and location provided by a user, can help the user to fetch the target data accurately and efficiently from the massive information on the Internet. Here, each client (i.e., each user interface) is considered to be trusted, since it is controlled by users completely. However, the server considered to be untrusted, since each server staff or attacker (who has hacked into the server) can access the user request data collected by the server. Therefore, it is a basic requirement for the privacy protection of personalized information retrieval how to accurately and efficiently obtain the target data from the server, without compromising user privacy.

Figure 1 shows a basic framework used by our method for user privacy protection in personalized information retrieval. As can be seen from Figure 1, each request R_0 submitted by a user generally contains a user identity U_0, a preference profile P_0, a point of interest T_0 and a location region L_0. It can be seen that user request data contains various kinds of user privacy (such as preference privacy, query privacy and location privacy), so the privacy protection method deployed at each client needs to prevent the untrusted server from analyzing and obtaining the user privacy. Note that most of personalized information retrieval platforms require users to login with real identities. Therefore, to not compromise the usability of personalized information retrieval, each dummy request constructed by the privacy method only changes three kinds of request data (i.e., interest points, preferences and locations), while retaining the user identity. As can be seen from Figure 1, the privacy protection method of personalized information retrieval is deployed at each client as a layer of middleware between the client and the server, whose data processing flows can be briefly described as follows.

S1: The dummy request construction component running on each client, by deeply mining the data distribution features and association features behind each user real service request R_0, and the historical distribution features after combining the historical service request sequence stored at each client, constructs a group of dummy requests $R_1, R_2, \ldots, R_n$, which are highly similar to R_0. Then, the dummy requests are submitted to the server together with the user request R_0 in a random order.

S2: The service result filtering component on a client, from the results $\mathcal{R}_0, \mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_n$ returned from the server (where $\mathcal{R}_k$ is a result corresponding to the request R_k), filters out the result $\mathcal{R}_0$ corresponding to the user real request R_0, while removing the other results corresponding to the dummy requests, and then returns $\mathcal{R}_0$ to the client user as the final result.

Figure 1. A basic framework used by our method for privacy protection in personalized information retrieval



Problem Statement

It can be seen from Figure 1 that the user privacy protection is transparent to both the server algorithm and the client user, so it requires no change to the existing architecture of a personalized information retrieval platform (it has good practical usability); the service result returned by the server is certainly a superset of the user actual result, so it requires no compromise to the accuracy of personalized information retrieval; and the loss of service performance caused by the introduction of user privacy protection is linearly related to the number of constructed dummy requests (it is linearly related to the intensity of user privacy protection), so the performance loss is controllable, which does not result in a significant impact on the efficiency of personalized information retrieval. Therefore, we conclude that a method based on the basic framework should be able to meet the expected requirements in terms of usability, accuracy and efficiency.

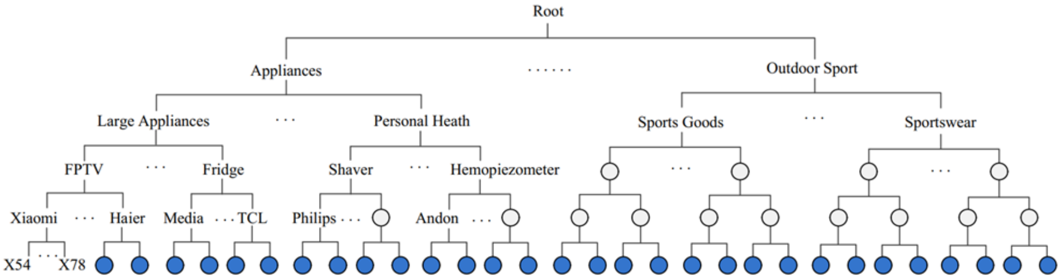
It can be seen from Figure 1 that the dummy requests generated from the dummy construction component play an important role in the expected security requirement, i.e., which are the key to the user privacy protection of personalized information retrieval. However, a user request sequence generally exhibits regular (non-random) distribution features. For example, a user often likes to search information around some fixed topics at some fixed location regions during a certain period of time. Therefore, it is easy for dummy requests that are randomly constructed to be ruled out by attackers based on the feature analysis. In order to improve the security of all kinds of user privacy on the untrusted server-side, the dummy construction component should ensure that its generated dummy requests are difficult to distinguish from the true ones, so as to mix up the user requests and then cover up the user privacy, i.e., ideal dummy requests should meet the following two constraints: (1) having highly-similar recognizable distribution features with the user requests, to mix up the user requests effectively and make it difficult for an attacker to make use of feature analysis to rule out the dummy requests; and (2) effectively reducing the prominence of all kinds of user privacy (such as query privacy, location privacy and preference privacy) on the untrusted server, to improve the cover-up effect of dummy requests on the user privacy.

However, there are various forms of recognizable distribution features behind the request data of personalized information retrieval, such as location frequency features (e.g., a user often likes to search around some fixed location regions in a certain period of time), query topic frequency features (e.g., a user often likes to search around some fixed topics during a certain period of time), and semantic association features (there are strong semantic correlations among the preference profile, location regions and query points of interest issued by the same user). In summary, it is a challenging task how to construct a group of ideal dummy requests for each user request, so as to achieve the expected goal of comprehensively improving the security of all kinds of user privacy in personalized information retrieval.

Assumption and Definition

In personalized information retrieval, each service request contains a preference profile, a user query (a query point of interest) and a location (Wang et al., 2021). In this subsection, we present some basic concepts and assumptions used in this study. First, to help describe related concepts, we introduce an external semantic knowledge repository, and assume that it can meet the following two requirements.

Figure 2. A sample tree structure of a topic hierarchy



Assumption 1: A knowledge repository is a large topic hierarchy, which is denoted by $\mathbb{H}$. It is assumed that the knowledge repository $\mathbb{H}$ can cover the whole topic domain of human knowledge, i.e., for any given recognizable human topic (e.g., a query topic, a preference or an interest point), we can find the corresponding node in the knowledge repository.

Assumption 2: It is assumed that the topic hierarchy of a knowledge repository is well organized, i.e., each leaf node of the knowledge repository has the same depth (the length of the path from each leaf node to the root node is the same to each other). In the knowledge repository, a leaf node denotes a query point of interest (which is considered to be a special topic without child nodes, i.e., whose hierarchical level is equal to 0), and a non-leaf node denotes a knowledge topic.

Figure 2 shows a sample tree structure of the topic hierarchy of a knowledge repository. Note that the depth of each leaf node to the root in the topic hierarchy of a human classification repository (such as the topic hierarchy of Chinese Library Classification) may be inconsistent with each other (Li et al., 2019). Below, based on the knowledge repository, we describe a preference profile and a user query (a query point of interest).

Definition 1 (Preference Profile): A preference profile (denoted by $\boldsymbol{P}$) is a hierarchical representation of topic preferences, which is a rooted subtree of the knowledge repository $\mathbb{H}$. Here, a rooted subtree refers to a subtree that retains the root node of the knowledge repository. In a preference profile $\boldsymbol{P}$, each node denotes a preference topic, which is associated with a preference score to denote the preference degree of a user to the topic. The preference score $PR\left(H,\boldsymbol{P}\right)$ associated with each node $H\in\boldsymbol{P}$ can be recursively calculated by those of all the leaf nodes (where $DH\left(H\right)$ represents the set of direct child nodes of a topic H).

$$PR\left(H,\ \boldsymbol{P}\right)=\sum\nolimits_{H'\in\ DH\left(H\right)}PR\left(H',\ \boldsymbol{P}\right)/\left|DH\left(H\right)\right|$$

Definition 2 (Interest Point): A user query (denoted by T) represents a query point of interest (e.g., Sinopec gas station). Here, we assume that each query interest point is chosen from the leaf nodes of the knowledge repository $\mathbb{H}$. Also, we assume that each user query is only associated with one query interest point. Below, we use $\mathbb{T}$ to denote the domain of interest points, which consists of all the leaf nodes from the repository $\mathbb{H}$.

A location region is a collection of location cells, i.e., the smallest location regions are the location cells themselves, and the largest location region is the location map. A location region is associated with a hierarchical level, and all the location regions at the same hierarchical level constitute a partition of the whole location map, i.e., they can divide the whole location map into several disjoint physical regions.

Definition 3 (Location Region): A location region is a set of location cells, which meet the following attributes. (1) A location region is associated with a hierarchical level, and the bigger the hierarchical level a location region has, the more location cells it contains (so the location map itself is a region of the biggest level, and each location is a region of the smallest level). (2) Any two regions at the same hierarchical level are not intersected with each other. (3) Any two regions at the same hierarchical level are of the same area size. (4) The union of all the regions at the same hierarchical level is equal to the map itself. (5) Each location region (except for the map) is contained in a region of a bigger hierarchical level.

In personalized information retrieval, except for a preference profile and a query point of interest, each service request also contains a query location (denoted by L) to denote a location cell where a user issues his service request. Now, combined with Defintions 1 and 2, we formulate a service request of personalized information retrieval, and its privacy.

Definition 4 (Request Privacy): Each service request (denoted by R) of personalized information retrieval consists of four tuples, represented by $R=U,T,P,L$, where U denotes a user identity, T a query interest point, P a preference profile, and L a query location. Then, the privacy related to query interest points is called query privacy, the privacy related to preference profiles is called preference privacy, and the privacy related to query locations is called location privacy. The three kinds of privacy constitute the main body of the protection of user privacy in personalized information retrieval.

Attack Model

It has been pointed out that the personalized information retrieval server is considered to be untrusted, since it is out of control of users, and it is the main target of attackers. Below, we introduce an attack model to define the attack ability of attackers in this study. First, we assume that the attackers have in advance obtained all the sequences of historical requests of personalized information retrieval submitted by the clients (i.e., a historical request sequence includes the real requests and the corresponding dummy requests, and a historical request sequence consists of a query subsequence, a location subsequence and a preference profile). Therefore, attackers can identify the user real requests by deeply analyzing the regular (non-random) distribution features (such as query features, location features, preference features and association features) specially shown by the real request sequences from users. Second, we assume that the attackers have obtained a complete topic hierarchy of the knowledge repository $\mathbb{H}$. Finally, we assume that the attackers have known the existence of the privacy protection algorithm deployed at each client, and obtained a copy of the algorithm in advance. Therefore, the attackers can input their mastered request sequences into the algorithm, and then analyze the output of the algorithm, to identify the user real requests.

Proposed Method

Based on the system model and the attack model, this section describes our method for privacy protection in personalized information retrieval, which includes a privacy model and its implementation algorithm. First, we define a privacy model to formulate the constraints that a group of dummy requests constructed by each client for a user request should meet. Second, with help of the knowledge repository, we design and present an implementation algorithm for the privacy model, so as to generate a group of dummy requests that meet the constraints defined in the privacy model.

Privacy Model

In the previous section, we have briefly discussed some problems that need to be taken into account for the construction of dummy requests. In this section, we further formulate the constraints that ideal dummy requests should satisfy, to provide a theoretical reference for the implementation of the dummy construction algorithm. First of all, we consider the distribution features of user query locations. Note that a user often likes to search around fixed location regions during a certain period of time, consequently, making a user query location sequence of a regular (non-random) distribution features. Therefore, this requires that dummy locations should be not randomly constructed, which should also reflect similar distribution features to the corresponding user locations; otherwise, an attacker can easily rule out the dummy locations according to the feature analysis. Table 1 present the main symbols used in the privacy model. Below, we introduce a concept of location region frequency vector defined on a location sequence to describe such regular distribution features.

Definition 5 (Location Region Frequency Vector): A location sequence is a time series of query locations issued by the same user over a period of time, denoted by $\boldsymbol{L}$. Then, for any given region D, the frequency of occurrence of a sequence $\boldsymbol{L}$ related to the region can be defined as $FR\left(D,\ \boldsymbol{L}\right) = \left|\left\{L_i \mid L_i \in \boldsymbol{L} \wedge L_i \in D\right\}\right|$. Let $\mathbb{D}_k$ denote all the location regions at the hierarchical level k. Then, we further define the location region frequency vector $FR_k\left(\boldsymbol{L}\right)$ of a location sequence $\boldsymbol{L}$ at the level k, which is an ordered sequence formed by the frequency of occurrence of the location sequence $\boldsymbol{L}$ related to each location region belonging to $\mathbb{D}_k$, i.e., $FR_k\left(\boldsymbol{L}\right) = \left[FR\left(D_i,\ \boldsymbol{L}\right) \mid D_i \in \mathbb{D}_k\right]$, where $FR\left(D_i,\ \boldsymbol{L}\right) \geq FR\left(D_{i+1},\ \boldsymbol{L}\right)$.

Table 1. Symbols and meanings

Symbols	Meanings
$\boldsymbol{L}$, $\boldsymbol{T}$, $\boldsymbol{P}$	A location sequence, a query sequence and a preference profile
$FR\left(D,\ \boldsymbol{L}\right)$	The frequency of occurrence of a location sequence $\boldsymbol{L}$ related to a region D
$\mathbf{sim}\left(\boldsymbol{L}_1,\ \boldsymbol{L}_2\right)$	The feature similarity of two location sequences $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$
$FR\left(H,\ \boldsymbol{T}\right)$	The frequency of occurrence of a query sequence $\boldsymbol{T}$ related to a topic H
$\mathbf{sim}\left(\boldsymbol{T}_1,\ \boldsymbol{T}_2\right)$	The feature similarity of two query sequences $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$
$\mathbf{sim}\left(\boldsymbol{P}_1,\ \boldsymbol{P}_2\right)$	The feature similarity of two preference profiles $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$
$RE\left(T,\ \boldsymbol{P}\right)$	The relevance of a query $T \in \boldsymbol{T}$ to the preference profile $\boldsymbol{P}$
$RE\left(T,\ L\right)$	The relevance of any query interest point $T \in \boldsymbol{T}$ to any location $L \in \boldsymbol{L}$
$\mathbf{sim}\left(\boldsymbol{R}_1,\ \boldsymbol{R}_2\right)$	The comprehensive feature similarity of two sequences $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$
$\mathbf{exp}\left(H,\ \mathbb{P}\right)$	The prominence of a topic H related to a preference profile $\mathbb{P}$
$\mathbf{exp}\left(H,\ \boldsymbol{T}\right)$	The prominence of a topic H related to a query sequence $\boldsymbol{T}$
$\mathbf{exp}\left(L_0,\ \boldsymbol{L}\right)$	The prominence of a location L_0 related to the sequence $\boldsymbol{L}$

Definition 6 (Location Feature Similarity): Let n_L denote the biggest hierarchical level of location regions (the level of the map itself), and the smallest hierarchical level of location regions is equal to 0 (the level of a location cell). Then, for any given two location sequences $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$, the feature similarity between them can be measured by the similarity between their location region frequency vectors at each level

$$\mathbf{sim}\left(\boldsymbol{L}_1,\ \boldsymbol{L}_2\right) = \sum\nolimits_{k=0}^{n_L} \alpha_k \cdot \boldsymbol{EJ}\left(FR_k\left(\boldsymbol{L}_1\right), FR_k\left(\boldsymbol{L}_2\right)\right)$$

where $\boldsymbol{EJ}$ denotes the Jaccard similarity between two vectors, and $\alpha_0 + \alpha_1 + \ldots + \alpha_{n_L} = 1$ and the value of α_k is preset in advance (their values are set to be equal to each other in the algorithm implementation).

Next, we consider the distribution features of user queries (interest points). Similar to location features, a user often likes to search around some fixed query topics (such as Food, Hospital and Hotel) during a certain period of time, such that the topics contained in a user query sequence show regular (non-random) distribution features. Obviously, to ensure the mix-up effect, a dummy query sequence should also reflect similar distribution features to their corresponding user query sequence. Below, we introduce a concept of query topic frequency vector defined on a query sequence to capture such non-random distribution features.

Definition 7 (Query Topic Frequency Vector): A query sequence is a time series of query interest points initiated by the same user over a period of time, denoted by $\boldsymbol{T}$. Then, for any query topic H, the frequency of occurrence of a query sequence $\boldsymbol{T}$ related to the topic can be defined as $FR\left(H,\ \boldsymbol{T}\right) = \left|\left\{T_i \mid\ T_i \in \boldsymbol{T} \wedge T_i \in H\right\}\right|$, where $T_i \in H$ denotes that a query interest point T_i is contained in a topic H. Let $\mathbb{H}_k$ denote all the query topics at the hierarchical level k. Then, we can further define a query topic frequency vector $FR_k\left(\boldsymbol{T}\right)$ of a query sequence $\boldsymbol{T}$ at the hierarchical level k, which is an ordered sequence composed of the frequency values of occurrence of the query sequence $\boldsymbol{T}$ related to each topic in $\mathbb{H}_k$, i.e., $FR_k\left(\boldsymbol{T}\right) = \left[FR\left(H_i,\ \boldsymbol{T}\right) \mid H_i \in \mathbb{H}_k\right]$, where $FR\left(H_i,\ \boldsymbol{T}\right) \geq FR\left(H_{i+1},\ \boldsymbol{T}\right)$.

Definition 8 (Query Feature Similarity): Let n_H denote the highest hierarchical level of the knowledge repository (i.e., the level of the root node), and the lowest level of the knowledge repository is equal to 0 (i.e., the level of a query interest point). Then, for any given two query sequences $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$, the query feature similarity between both can be measured by the similarity between their query topic frequency vectors at each hierarchical level, i.e.,

$$\mathbf{sim}\left(\boldsymbol{T}_1,\ \boldsymbol{T}_2\right) = \sum\nolimits_{k=0}^{n_H} \beta_k \cdot \boldsymbol{EJ}\left(FR_k\left(\boldsymbol{T}_1\right), FR_k\left(\boldsymbol{T}_2\right)\right)$$

where $\beta_0 + \beta_1 + \ldots + \beta_{n_H} = 1$ and the value of each parameter β_k is preset by the system in advance (in the algorithm implementation, their values are simply set to be equal to each other).

A preference profile is an important input of personalized information retrieval. According to Definition 1, we can know that a preference profile is the rooted subtree of topic hierarchy of the knowledge repository, which is used to represent the personal interests and preferences of some user. Each node in a profile represents a preferred topic, and its associated preference score represents the preference degree of a user to the topic. In general, a preference profile is automatically constructed and generated by the system according to the online behavior records of some user. Due to the regularity of user online behaviors, a profile also shows a regular distribution features (the preference score of each node is non-random). Below, we introduce a concept of preference topic frequency vector of a profile to capture such regular distribution features.

Definition 9 (Preference Topic Frequency Vector): Let $\mathbb{H}_k$ denote all the topics at the hierarchical level k. Then, a preference topic frequency vector $PR_k\left(\boldsymbol{P}\right)$ of a preference profile $\boldsymbol{P}$ at the level k canbedefinedas $PR_k\left(\boldsymbol{P}\right) = \left[PR\left(H_i,\ \boldsymbol{P}\right) \mid H_i \in \mathbb{H}_k\right]$,where $PR\left(H_i,\ \boldsymbol{P}\right) \geq PR\left(H_{i+1},\ \boldsymbol{P}\right)$.

Definition 10 (Preference Feature Similarity): Let n_H denote the highest level of the knowledge repository (the level of the root node). Then, for any given two preference profiles $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$, the preference feature similarity between them can be measured by the similarity between their preference topic frequency vectors at each level, i.e.,

$$\mathbf{sim}\left(\boldsymbol{P}_1,\ \boldsymbol{P}_2\right) = \sum\nolimits_{k=1}^{n_H} \gamma_k \cdot \boldsymbol{EJ}\left(PR_k\left(\boldsymbol{P}_1\right), PR_k\left(\boldsymbol{P}_2\right)\right)$$

where $\gamma_1 + \gamma_2 + \ldots + \gamma_{n_H} = 1$, and the value of each parameter γ_k is preset in advance (in the algorithm implementation, their values are simply set to be equal to each other).

From Definition 4, we know that each request of personalized information retrieval issued by a user includes a query interest point, a query location and a preference profile; and thus a sequence of service requests issued by the same user during a period of time includes a query subsequence, a location subsequence and a preference profile. Also, there are association features among various kinds of subsequences, which are different from the distribution features of a sequence with single kind mentioned above. Therefore, the association features of request sequences refer to the semantic associations among various kinds of request data initiated by the same user over a period of time. For example, if a user has a preference topic Food, he/she may frequently search some information related to the interest point Food during a period of time; and if a user is near the hospital, he/she is likely to search hospital related information. Also, we note that such association features mainly exists between query interest points and preference profiles, or between query interest points and query locations. Below, we introduce a concept of query preference association vectors (Definition 11) and a concept of query location association vectors (Definition 12) to capture such association features among various kinds of request data.

Definition 11 (Query Preference Association Vector): A request sequence of personalized information retrieval is a time series of service requests (each request contains a user identify, a query interest point, a preference profile and a query location) issued by the same user during a period of time, which is denoted by $\boldsymbol{R}$. Let $\boldsymbol{P}$ and $\boldsymbol{T}$ denote the preference profile and the query sequence associated with $\boldsymbol{R}$, respectively. Then, the relevance of a query $T \in \boldsymbol{T}$ to the preference profile $\boldsymbol{P}$ can be defined as follows

$$RE\left(T,\ \boldsymbol{P}\right) = PR\left(H^*,\ \boldsymbol{P}\right) / \, HE\left(H^*\right),\ \text{where } H^* = \mathbf{arg}\min\nolimits_H HE\left(H\right)\ \textbf{s.t.}\ \ H \in \boldsymbol{P} \wedge T \in H$$

, where H^* denotes a topic contained in $\boldsymbol{P}$, which is also required to contain the query interest point T and have the lowest hierarchical level, and $HE\left(H^*\right)$ denotes the hierarchical level of H^*. Then, we can further define a query preference association vector $RT\left(\boldsymbol{R}\right)$ of the request sequence $\boldsymbol{R}$, which is an ordered sequence composed of the relevance of each query interest point in the query sequence $\boldsymbol{T}$ to the preference profile $\boldsymbol{P}$, so can be represented as $RT\left(\boldsymbol{R}\right) = \left[RE\left(T_i,\ \boldsymbol{P}\right) \mid T_i \in \boldsymbol{T}\right]$.

Definition 12 (Query Location Association Vector): Let $\boldsymbol{L}$ and $\boldsymbol{T}$ respectively denote a location sequence and a query sequence associated with a request sequence $\boldsymbol{R}$. Then, the relevance of any query interest point $T \in \boldsymbol{T}$ to any $L \in \boldsymbol{L}$ can be measured by whether T is matched with L (which is assigned in advance)

$$RE\left(T,\ L\right) = \begin{cases} 1 & if\, T\, is\, mathed\, with\, L \\ 0 & otherwise \end{cases}$$

Then, we can further define a query location association vector of the request sequence $\boldsymbol{R}$. It is an ordered sequence composed of the relevance of each query interest point in the query sequence $\boldsymbol{T}$ to the corresponding location in the location sequence $\boldsymbol{L}$, i.e., $RL(\boldsymbol{R}) = [RE(T,\ L)\ |T, L \in \boldsymbol{T}, \boldsymbol{L}]$.

Now, based on Definition 5 and Definition 6 (location features), Definition 7 and Definition 8 (query features), Definition 9 and Definition 10 (preference features), and Definition 11 and Definition 12 (association features), the comprehensive feature similarity between two request sequences of personalized information retrieval can be further defined to formulate the mix-up effect of a request sequence to another.

Definition 13 (Sequence Feature Similarity): Let $\boldsymbol{R}$ denote a request sequence of personalized information retrieval. Let $\boldsymbol{P}$, $\boldsymbol{T}$ and $\boldsymbol{L}$ denote the preference profile, the query sequence and the location sequence contained in $\boldsymbol{R}$, respectively. Then, for any given two sequences $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$, the comprehensive feature similarity between them can be measured as follows

$$\mathbf{sim}(\boldsymbol{R}_1,\ \boldsymbol{R}_2) = (\theta_1 \cdot \mathbf{sim}(RT(\boldsymbol{R}_1), RT(\boldsymbol{R}_2)) + \theta_2 \cdot \mathbf{sim}(RL(\boldsymbol{R}_1), RL(\boldsymbol{R}_2))) \cdot (\omega_1 \cdot \mathbf{sim}(\boldsymbol{P}_1,\ \boldsymbol{P}_2) + \omega_2 \cdot \mathbf{sim}(\boldsymbol{T}_1,\ \boldsymbol{T}_2) + \omega_3 \cdot \mathbf{sim}(\boldsymbol{L}_1,\ \boldsymbol{L}_2))$$

where $\theta_1 + \theta_2 = 1$ and $\omega_1 + \omega_2 + \omega_3 = 1$, whose values are preset in advance.

From the attack model, we know that when an attacker cannot identify the user real requests in advance, he/she can only guess the user possible interest preferences, as well as the user possible locations by analyzing the user historical request records (including the query sequences, the location sequences, and preference profiles). As a result, the more prominent a user topic in the historical request sequences, the more likely an attacker can successfully guess the user topic. Therefore, we introduce a concept of topic prominence to describe the security of query privacy and preference privacy on the untrusted server.

Definition 14 (Preference Topic Prominence): Let denote all the topics at the hierarchical level . Then, for any topic defined at the level , the prominence of the topic related to a preference profile $\mathbb{P}$ can be defined as follows

$$\mathbf{exp}(H,\ \mathbb{P}) = \sum\nolimits_{\boldsymbol{P} \in \mathbb{P}} PR(H,\ \boldsymbol{P}) \cdot \left(\sum\nolimits_{\boldsymbol{P} \in \mathbb{P}} \sum\nolimits_{H' \in \mathbb{H}_k} PR(H',\ \boldsymbol{P})\right)^{-1}$$

Definition 15 (Query Topic Prominence): Let $\mathbb{H}_k$ denote all the topics at the hierarchical level k. Then, for any topic H defined at the level k, the prominence of H related to a query sequence $\boldsymbol{T}$ can be defined as

$$\mathbf{exp}(H,\ \boldsymbol{T}) = \sum\nolimits_{\boldsymbol{T} \in \mathbb{T}} FR(H,\ \boldsymbol{T}) \cdot \left(\sum\nolimits_{\boldsymbol{T} \in \mathbb{T}} \sum\nolimits_{H' \in \mathbb{H}_k} FR(H',\ \boldsymbol{T})\right)^{-1}$$

In addition, from the system model, we know that in order to protect the location privacy, the dummy locations should keep a safe distance from the corresponding user locations. Therefore, based on such a requirement, we introduce a concept of location prominence to describe the security of location privacy (Definition 16) on the untrusted server.

Definition 16 (Location Prominence): For any location L_0 and any location set $\boldsymbol{L}$, the prominence of L_0 related to the sequence $\boldsymbol{L}$ can be defined as follows

$$\mathbf{exp}\left(L_0,\ \boldsymbol{L}\right) = \frac{1}{\left|\boldsymbol{L}\right|}\left|\left\{L_i \mid \left|L_i - L_0\right| \le \varepsilon \wedge L_i \in \boldsymbol{L}\right\}\right|$$

, where $\left|L_i - L_0\right|$ denotes the distance between two locations, and the threshold ε denotes a safe distance without disclosing the location privacy.

Now, based on Definition 13, Definition 14, Definition 15 and Definition 16, below, we further define a privacy model to formulate the constraints that ideal dummy requests should satisfy.

Definition 17 (Privacy Protection): For any given user request sequence $\boldsymbol{R}_0$ of personalized information retrieval (which includes a preference profile $\boldsymbol{P}_0$, a user query sequence $\boldsymbol{T}_0$ and a user location sequence $\boldsymbol{L}_0$), and a set of corresponding dummy request sequences $\mathbb{R}^*$ (Let $\mathbb{P}^*$, $\mathbb{T}^*$ and $\mathbb{L}^*$ denote the set of dummy preference profiles, the set of dummy query sequences, and the set of dummy location sequences associated with $\mathbb{R}^*$, respectively), then if the set $\mathbb{R}^*$ can meet the following constraints, it is deemed that the set $\mathbb{R}^*$ can comprehensively improve the security of all kinds of user privacy behind the user sequence $\boldsymbol{R}_0$.

(1) For each dummy request sequence in $\boldsymbol{R}_k$, its comprehensive feature similarity related to $\boldsymbol{R}_0$ should be greater than a given threshold, i.e., $\min_{\boldsymbol{R}_k \in \mathbb{R}^*} \mathbf{sim}\left(\boldsymbol{R}_k,\ \boldsymbol{R}_0\right) \ge \pi$, where π is a preset threshold. This constraint ensures that each dummy sequence is highly similar to the user sequence, which makes them difficult to be ruled out by an attacker according to feature analysis, to achieve a good cover-up effect.

(2) The set of dummy query sequences $\mathbb{T}^*$ can effectively reduce the prominence of each topic in the given user sensitive topic set $\mathbb{H}^*$, i.e., $\max_{H^* \in \mathbb{H}^*} \mathbf{exp}\left(H^*,\ \left\{\boldsymbol{T}_0\right\}\right) / \mathbf{exp}\left(H^*,\ \left\{\boldsymbol{T}_0\right\} \cup \mathbb{T}^*\right) \le \rho$, where ρ is a preset threshold. This constraint makes it difficult for an attacker to analyze the user sensitive topics directly from the dummy query sequences, to improve the security of query privacy.

(3) The set of dummy preference profiles $\mathbb{P}^*$ can effectively reduce the prominence of each topic in the given user sensitive preference set $\mathbb{H}^*$, i.e., $\max_{H^* \in \mathbb{H}^*} \mathbf{exp}\left(H^*,\ \left\{\boldsymbol{P}_0\right\}\right) / \mathbf{exp}\left(H^*,\ \left\{\boldsymbol{P}_0\right\} \cup \mathbb{P}^*\right) \le \rho$. This constraint makes it difficult for an attacker to analyze the user sensitive preferences directly from the dummy preference profiles, so as to improve the security of preference privacy.

(4) The set of dummy location sequences $\mathbb{L}^*$ can effectively reduce the prominence of each user location, i.e., $\max_{L_0 \in \boldsymbol{L}_0} \mathbf{exp}(L_0,\ \left\{L_0\right\} \cup \{$ the dummy locations in $\mathbb{L}^*$ corresponding to $L_0\}) \le \rho$. It ensures that the dummy location can well cover up the location privacy, to improve the security of location privacy.

Algorithm Implementation

In this section, we discuss the specific algorithm implementation of the privacy model, which can be divided into two steps. First, we discuss how to generate an ideal dummy request (including a dummy location, a dummy query interest point and a dummy profile) for a user current request by combining the user historical request sequence and the corresponding historical dummy sequence. Then, we discuss how to generate a group of dummy requests for a user current request, so that the set of final dummy request sequences can comprehensively improve the security of all kinds of user privacy in personalized information retrieval (i.e., the constraints in Definition 17).

According to the constraints of Definition 17, Step 1 aims to generate a dummy request meeting the following two conditions as much as possible for the user current request. (1) The location region frequency features, query topic frequency features, preference topic frequency features, and semantic association features of dummy requests are similar to those of the user requests to achieve a good mix-up effect. (2) The dummy location is of a safe distance from the user location, and the dummy profile and the dummy query are both independent of each user sensitive preference topic to achieve a good cover-up effect on all kinds of user privacy. Algorithm 1 presents a solution. It can be seen that Algorithm 1 uses a Greedy strategy, i.e., we do not try to obtain the optimal solution for Step 1 (the solution of the best match degree with the constraints of Definition 17), but try to obtain a solution meeting the constraints as much as possible.

In Algorithm 1, since the user profile $\boldsymbol{P}$ is almost unchanged during a period of time, we reconstruct a new dummy profile (Lines 1 to 4), only when the current user profile has been changed compared to the historical user profile. Next, Algorithm 1 constructs a corresponding dummy query point of interest (Lines 5 to 10) for the current user query. In this process, it is required that the dummy query topic should be semantically unrelated to each user sensitive topic (Line 7), and the obtained current dummy sequence should have highly similar query topic frequency features and association features with the user sequence (Line 8). Finally, Algorithm 1 constructs a corresponding dummy query location (Lines 11 to 16) for the user location. In this process, it is required that the dummy location should keep a safe distance from the user location (Line 13), and the new dummy location sequence after adding the current obtained dummy location should have highly similar location region frequency features and query location association features with the user sequence (Line 14). In Algorithm 1, if the set of dummy candidates that satisfy the given conditions cannot be obtained (if the loop condition of Line 6 or Line 12 cannot be satisfied), we would relax the condition constraint continuously (by the third expressions of Lines 6 and 12) until we can obtain a valid set of dummy candidates. Next, Lines 10 and 16 randomly select a corresponding dummy query and dummy location from the obtained candidate dummy set for the user request. Also, it can be seen that the output of Algorithm 1 is uncertain, i.e., for the same input, different runs output different results, because some random operations are used in Lines 10 and 16, as well as in Line 6 of the user-defined function, to better improve the security (see the analysis later). To facilitate the description of the algorithm, we assume that there exists at least one solution for Algorithm 1 (the loop conditions of Lines 6 and 12 cannot be always false).

```
Algorithm 1: Construction of a Dummy Request (with a Non-empty Historical Sequence)
Input: A current request $R = (\boldsymbol{P}, T, L)$; The knowledge repository $\mathbb{H}$; A set of sensitive topics $\mathbf{H}^*$; A historical user sequence $\boldsymbol{R} = (\boldsymbol{P}, \boldsymbol{T}, \boldsymbol{L})$; A historical dummy sequence $\boldsymbol{R}^* = (\boldsymbol{P}^*, \boldsymbol{T}^*, \boldsymbol{L}^*)$
Output: A dummy request $R^{\#} = (\boldsymbol{P}^{\#}, T^{\#}, L^{\#})$ corresponding to the current request $R$
01 IF The user current profile is changed compared to the historical profile THEN
```

```
01  Obtain the root node $H$ of the repository $\mathbb{H}$ (also the root of the user profile $\boldsymbol{P}$)
02  $\boldsymbol{P}^{\#} \leftarrow \{H\} +$ SEARCH ($\boldsymbol{P}$, $\mathbb{H}$, $\mathbb{H}^{*}$, $H$, $H$)
03 END IF
04 According to the similarity threshold and security threshold, set $d_1$ and $d_2$ to smaller values
05 FOR $\mathbf{T}^{*} \leftarrow \varnothing$; If the size of $\mathbf{T}^{*}$ is smaller; $d_1 \leftarrow 2d_1$ and $d_2 \leftarrow 2d_2$ DO
06   $\mathbf{T}^{*} \leftarrow \mathbb{T} - \{T_i \mid T_i \in H \wedge H \in \mathbf{H}^{*}\}$
07   $\mathbf{T}^{*} \leftarrow \{T^{*} \mid T^{*} \in \mathbf{T}^{*} \wedge sim(\boldsymbol{T} + T,\ \boldsymbol{T} + T^{*}) \le d_1 \wedge |RE(T^{*}, \boldsymbol{P}^{\#}) - RE(T, \boldsymbol{P})| \le d_2\}$
08 END FOR
09 From the set $\mathbf{T}^{*}$, randomly select an interest points to construct a dummy query $T^{\#}$
10 According to the similarity threshold and security threshold, set $d_1$ and $d_2$ to smaller values
11 FOR $\mathbf{L}^{*} \leftarrow \varnothing$; If the size of $\mathbf{L}^{*}$ is smaller; $d_1 \leftarrow 2d_1$ and $d_2 \leftarrow 2d_2$ DO
12   $\mathbf{L}^{*} \leftarrow \mathbb{L} - \{L_i \mid |L_i - L| \rangle \varepsilon\}$
13   $\mathbf{L}^{*} \leftarrow \{L^{*} \mid L^{*} \in \mathbf{L}^{*} \wedge \mathbf{sim}(\boldsymbol{L} + L,\ \boldsymbol{L} + L^{*}) \le d_1 \wedge |RL(T^{\#}, L^{*}) - RL(T, L)| \le d_2\}$
14 END FOR
15 From the set $\mathbf{L}^{*}$, randomly select a dummy location $L^{\#}$; RETURN $R^{\#} = (\boldsymbol{P}^{\#}, T^{\#}, L^{\#})$

SEARCH (A user profile $\boldsymbol{P}$; $\mathbb{H}$; $\mathbf{H}^{*}$; A user current node $H$; A dummy topic node $H^{\#}$)
01 IF the current node $H$ is a leaf node of the user profile $\boldsymbol{P}$ THEN
02  Set the preference score for $H^{\#}$, i.e., $PR(H^{\#}) \leftarrow PR(H, \boldsymbol{P})$; RETURN $H^{\#}$
03 ELSE
04  Obtain the set of child nodes of $H$ related to the user profile $\boldsymbol{P}$, denoted by $\mathbf{H}$
05  Obtain the set of child nodes of $H^{\#}$ related to the knowledge repository $\mathbb{H}$, denoted by $\mathbf{H}'$
06   $\mathbf{H}' \leftarrow \mathbf{H}' - \mathbf{H}^{*}$; From $\mathbf{H}'$, randomly select a child set $\mathbf{H}^{\#}$ of the same size to $\mathbf{H}$
07  FOREACH $H_1, H_2 \in \mathbf{H}, \mathbf{H}^{\#}$ DO SEARCH ($\boldsymbol{P}$, $\mathbb{H}$, $\mathbf{H}^{*}$, $H_1$, $H_2$) END FOR
08 END IF
```

In addition, a user-defined function (SEARCH) is used in the algorithm. In the SEARCH function, if the user node processed currently is a leaf node of the profile, we do not continue to recursively call the function, but assign the dummy leaf node with the same preference score as the corresponding user leaf node (Line 2); otherwise, we randomly search and obtain a dummy topic semantically unrelated to each user sensitive topic (Lines 4 to 6) and recursively processes the next level topics belonging to the dummy topic (Line 7). Finally, the function can ensure that the final generated dummy profile has the same preference topic distribution features with the user profile. It can be seen that the time complexity of Algorithm 1 is equal to $\mathbf{O}(|\boldsymbol{P}| + |\boldsymbol{T}| + |\boldsymbol{L}|)$.

It can be seen that in Step 1, we only generate a dummy request for the user current request. Therefore, in the end, we briefly discuss how to generate a group of dummy requests for the current user request. This problem can be solved by running Algorithm 1 several times, whose number is greater than or equal to the security threshold A. After combined with the algorithm, it can be seen that the final obtained set of dummy sequences may not meet the constraints of Definition 17. However, in Algorithm 1, we make each dummy request to as much as possible meet Definition 6 (location feature similarity), Definition 8 (query feature similarity), Definition 10 (preference feature similarity) and Definition 13 (association feature similarity), as well as the privacy requirements mentioned in Definition 17 (query privacy, location privacy and preference privacy), to generate a group of ideal dummy request sequences for the user request sequence. In fact, the experimental results presented later also show that the set of dummy request sequences obtained by our algorithm can well meet the similarity constraint and the privacy constraint mentioned in Definition 17, and the running number of Algorithm 1 is generally equal to the security threshold A, in each construction of the dummy request set.

EXPERIMENTS

Experimental Setup

Reference dataset. First, we introduce the construction of the knowledge repository. We use a similar reference dataset used in (Wu et al., 2018b), i.e., based on the product classification catalog of Jingdong Mall (one of the most famous e-commerce platforms in the world), we construct a six-level classification topic tree, which includes 20,751 non-leaf topic nodes and 198,410 leaf product nodes (i.e., query interest points). To simplify the structure, each topic at Level 1 contains only 10 leaf nodes. In addition, the topic tree has been optimized in advance (such as sorting all products and all topics at the same level), so that in Algorithm 1, we can quickly obtain a candidate set of dummy topics or a candidate set of dummy products. Second, we introduce the construction of location data. We use a similar reference dataset used in (Wu et al., 2020b), i.e., the map is extracted from a square area with the size of 80 square kilometers in Connecticut. We divide the whole map into 80000^2 location cells, and divide all the regions into five levels (80000^2 location cells at Level 0; 800^2 regions at Level 1; 200^2 regions at Level 2; 50^2 regions at Level 3; and the map itself at Level 4). Finally, to in advance establish the match degree between locations and topics, we divide all the location cells into 20 categories, and then randomly set the match degree (1 or 0) between each topic at Level 1 and each location category, consequently, obtaining a two-dimensional 0-1 matrix from the location categories to the query topics, so that given any location cell, we can know all the topics it can support according to the location category it belongs to.

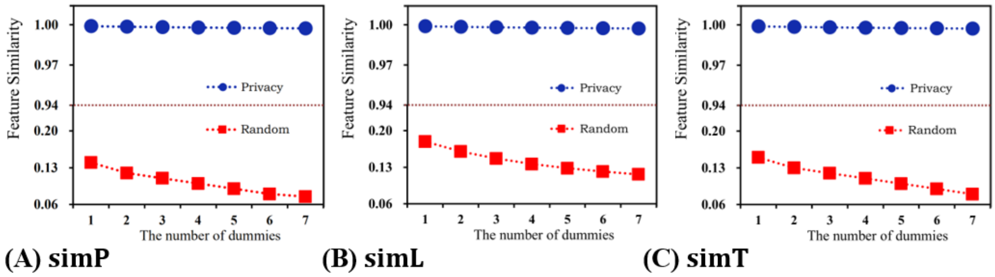
User request sequences. To obtain user request sequences, we need to construct user query sequences and their corresponding user location sequences, as well as user preference profiles (which are assumed to remain unchanged). First, each preference profile is constructed on the classification topic tree, according to the topic normal distribution, where the number of leaf nodes, the number of topic nodes, and the preference score of leaf nodes are all adjustable parameters. Second, each user query sequence is constructed on top of 198,410 possible products, according to the topic normal distribution. Finally, we adopt a famous road network data generator (Chatzikokolakis et al., 2015) to construct each user query location sequence, to make the location distribution more accordant to the real situation. In addition, the length (the number of requests) of each user request sequence is also an adjustable parameter.

Candidate algorithms. In the experimental evaluations, our method (named by Privacy) is compared with a random method (named by Random). In the random method, each leaf node and each topic node in a dummy preference profile are randomly constructed based on the classified topic tree, but the number of nodes in the dummy profile is consistent with that of the user profile. In addition, each dummy query sequence is randomly constructed based on the product space, and each dummy location sequence is randomly constructed based on the location cell space, but the length of each of the two sequences is consistent with that of the user request sequence. In the experiments, we do not compare our method with other methods in the literature review section, because all the methods are proposed under different privacy models, and they generally are targeted only for a single kind of privacy data (such as location data).

Experimental Result

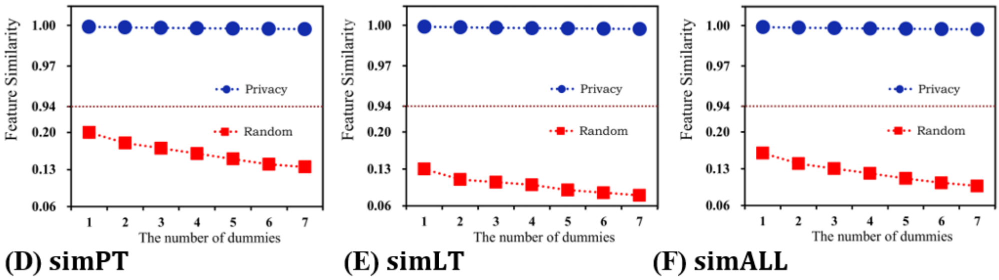
The first group of experiments aims to evaluate the feature similarity of the dummy request sequences generated by our method to the user real request sequences, i.e., whether the dummy request sequences can achieve a good mix-up effect on the user request sequences. Here, based on Definitions 6, 8, 10 and 13, we introduce six feature similarity metrics, i.e., preference feature similarity (denoted by $\mathbf{simP}$), location feature similarity (denoted by $\mathbf{simL}$), query feature similarity (denoted by $\mathbf{simT}$) and preference association feature similarity (denoted by $\mathbf{simPT}$), location association feature similarity (denoted by $\mathbf{simLT}$) and comprehensive feature similarity (denoted by $\mathbf{simAll}$). It is obvious that the larger values the metrics have, the better the mix-up effect. In the experiment, the length of a user request sequence is fixed at 1000, and the number of dummy request sequences constructed for a user sequence is set from 1 to 7. The experimental results are shown in Figure 3, where the caption of each subgraph denotes the feature similarity metric, the X-axis denotes the number of dummy sequences constructed for a user sequence, and the Y-axis denotes the similarity metric values.

Figure 3a. The feature similarity evaluation



(A) simP (B) simL (C) simT

Figure 3b. The feature similarity evaluation

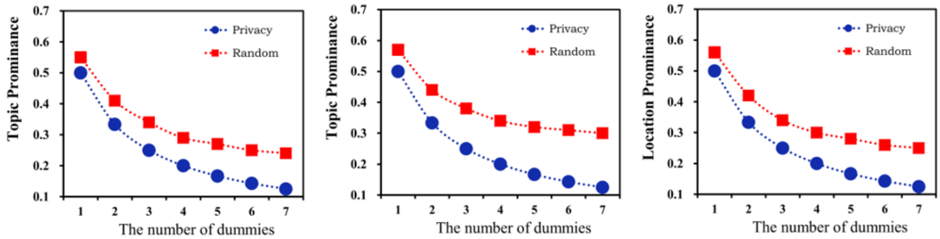


(D) simPT (E) simLT (F) simALL

From the subgraphs (A) to (C), we see that compared with the random method, the dummy request sequences constructed by our method has much more similar single distribution features to the user sequences (including preference features, location features and query features). Specifically, for the dummy sequences constructed by the random method, their single feature similarity values are less than 0.2, because the distribution features of these dummy sequences are basically uniform, which are quite different from those of the user sequences (that are usually issued around some fixed regions or topics). However, for the dummy sequences constructed by our method, their single feature similarity values are all close to 1.0, and would not change significantly with the increase of the number of constructed dummy requests. From the subgraphs (D) to (E), we see that compared with the random method, each dummy request sequence constructed by our method has much better association feature similarity with the user request sequence (close to 1, while close to 0 for the random method), and would be nearly unchanged with the increase of the number of constructed dummy requests. This is because the random method does not consider the association features among various kinds of request data when constructing dummy requests, however, which have been taken into full consideration by our method. Finally, from the subgraph (F), it can be seen that our method not only fully considers the query features, location features and preference features, but also takes into account the association features among various kinds of request data, making that the final obtained dummy sequences have good comprehensive feature similarity values (close to 1) with the user sequence. In summary, the dummy sequences constructed by our method have highly consistent feature similarity (including preference features, location features, query features and association features) to the user sequence, making it difficult for an attacker to rule out the dummy requests generated by the privacy method according to feature analysis, and achieving a good mix-up effect on the user request sequences.

The second group of experiments aims to evaluate the cover-up effect of the dummy request sequences generated by our method on the location privacy, query privacy and preference privacy, i.e., whether the dummy requests can effectively reduce the prominence of user privacy on the untrusted server. Here, we introduce some metrics of privacy prominence, which are mainly constructed based on Definitions 14, 15 and 16, specifically including location privacy prominence (denoted by **expL**), query privacy prominence (denoted by **expT**) and preference privacy prominence (denoted by **expP**). Obviously, the smaller values the metrics, the better the cover-up effect of dummy request sequences on all kinds of user privacy, which means the more difficult for an attacker to analyze out the user privacy. In the experiments, the length of a user request sequence is fixed to 1000, and the number of dummy request sequences constructed for a user sequence is set from 1 to 7. The experimental results are shown in Figure 4.

Figure 4. The privacy prominence evaluation



(A) expP (B) expL (C) expT

From Figure 4, it can be seen that the dummy request sequences constructed by our method can effectively reduce the prominence of location privacy, query privacy and preference privacy. Moreover, the degree of improvement of privacy prominence is basically positively related to the number of dummy requests constructed for a user request (linearly related to the number of dummies), and would not change significantly with the changes of the length of a user request sequence, the size of a user preference profile and other factor. However, compared to our method, although the dummy sequence constructed by the random method can also reduce the prominence of all kinds of user privacy, its stability is relatively poor, and it is not linearly related to the number of dummies. This is because when constructing a dummy profile (or query), it is possible for the random method to select a dummy from a user sensitive topic, and when constructing a dummy location, it is also possible to select a dummy of an unsafe distance from the user location, so that the cover-up effect of dummy requests on the user privacy cannot be fully exploited. However, more importantly, the first group of experimental results shows that the dummy sequences constructed by the random method have poor feature similarity to the user sequence, consequently, making them easy to be ruled out by an attacker according to the feature analysis, and in turn, losing the cover-up effect on the user privacy. To sum up, the dummy request sequences constructed by our method can significantly decrease the prominence of users' query privacy, preference privacy and location privacy, i.e., they can achieve a good cover-up effect on all kinds of user privacy, which makes is difficult for an attacker to directly guess out the user privacy, under the premises of not finding out the user real requests.

Analysis and Discussion

To ensure the practicality of an existing personalized information retrieval platform, privacy protection not only should be transparent to the server algorithm and each client interface (i.e., the usability), but also should not compromise the accuracy and efficiency of information retrieval, so an ideal method of privacy protection should meet the actual needs of personalized information retrieval in terms of usability, accuracy, efficiency and security. In this section, we first formulate the constraints that an ideal privacy protection method should meet, and then analyze and discuss the effectiveness of the proposed method.

Constraint 1 (Accuracy): The result $\mathcal{R}_0^*$ returned from the server of personalized information retrieval should be a superset of the real result $\mathcal{R}_0$ corresponding to the original user request R_0, i.e., $\mathcal{R}_0^* \supseteq \mathcal{R}_0$. This constraint ensures that the user can finally obtain the target result (i.e., the accuracy).

Constraint 2 (Efficiency): The size $\left|\mathcal{R}_0^*\right|$ of the result returned from the server should be linearly related to the size $\left|\mathcal{R}_0\right|$ of the result real corresponding to the user request R_0, i.e., $\left|\mathcal{R}_0^*\right| \propto \left|\mathcal{R}_0\right|$ or $\left|\mathcal{R}_0^*\right| \le \varphi\left|\mathcal{R}_0\right|$, where φ is a positive real number less than a given threshold (denoting the degree of efficiency loss that the user can accept).

Constraint 3 (Usability): The usability of a privacy method for personalized information retrieval includes two levels: (1) Level I, i.e., not changing the usage pattern of each client user; and (2) Level II, i.e., not changing the retrieval algorithm on the server.

Constraint 4 (Security I): According to a single kind of request data submitted by a user in an information retrieval activity, it should be difficult for the server to guess out the user privacy accurately. Let $Pr\left(T\right)$ denote the probability of successfully guessing the user privacy, and $R_0^* = \left(U_0^*, T_0^*, P_0^*, L_0^*\right)$ denote a new request (instead of the original request) to be submitted to the server. Then, we have that $Pr\left(T_0^*\right) \le 1 / \mu; Pr\left(P_0^*\right) \le 1 / \mu; Pr\left(L_0^*\right) \le 1 / \mu$, where μ denotes the strength of privacy protection.

Constraint 5 (Security II): According to various kinds of request data submitted by a user in an information retrieval activity, it is difficult for the server to accurately guess out the user privacy, i.e., we have that $Pr\left(R_0^*\right) = Pr\left(T_0^* P_0^* L_0^*\right) \leq 1 / \mu$.

Constraint 6 (Security III): It is difficult for the server to accurately guess out the user privacy by analyzing various types of request data submitted by the same user in multiple retrieval activities during a certain period of time. Let $R_1^* R_2^* \ldots R_n^*$ denotes a new request sequence (instead of the user request sequence) to be submitted to the server. Then, we have that $Pr\left(R_1^* R_2^* \ldots R_n^*\right) \leq 1 / \mu$.

According to the constraints mentioned above, we briefly analyze the effectiveness of the proposed method for the protection of user privacy in personalized information retrieval.

Observation 1: Our method can meet the accuracy (Constraint 1), efficiency (Constraint 2) and usability (Constraint 3) constraints related to the protection of user privacy in personalized information retrieval.

Explain: Since our method is proposed based on the system model presented in Figure 1, it can be easily proved according to the advantages of the system model mentioned previously.

Observation 2: Our method can meet the three-level security constraints (i.e., Constraints 4 to 6) related to the protection of user privacy in personalized information retrieval.

Explain: In our method, although each user request has been mixed up within a group of dummy requests, due to the strong feature associations among the requests from the same sequence, it is still possible for an attacker to divide the request records collected by the server from clients into several independent sequences (to obtain $R_0, R_1, R_2, \ldots, R_n$). Next, we take into account two cases, i.e., (1) the possibility of an attacker to find out the user sequence from all the sequences; and (2) under the premise of not finding the user sequence, the possibility of all kinds of user privacy can be directly analyzed out from the sequences.

(1) At this time, the attacker can analyze the user sequence according to the prior knowledge that each user sequence shows regular (non-random) distribution features. However, each dummy request sequence constructed by our method shows highly similar recognizable distribution features (including single distribution features and association distribution features) with the user sequence. As a result, it is difficult for the attacker to distinguish the user requests from the dummy ones based on the request data formed in one retrieval activity, or based on the request sequences formed in multiple activities within a certain period of time. In other words, the probability of successfully finding out the user sequence is equal to 1/(n+1). In addition, it should be noted that the attacker has also obtained a copy of the privacy algorithm. At this time, for a collection of requests submitted by a client in a retrieval activity (which contains a user request and dummy requests), the attacker can input each request into the algorithm one by one, and then observe whether the algorithm can output the remaining requests. If successful, it indicates that the current request is from a user, and in turn, the attacker can obtain the user sequence. However, such an attempt cannot succeed, because in the algorithm implementation, each dummy is randomly selected from a large set of candidates (see Algorithm 1), i.e., each time the same data is input, different result may be output.

(2) At this time, since the user request sequence is not determined in advance, the attacker can only analyze all the related preference topics (query topics, or query locations) behind the mastered whole sequence set (n), and then guess which topics are user-sensitive one by one. However, since the prominence of each sensitive preference topic in the sequence set has been significantly reduced compared with that in the user sequence (see the experimental result), the probability of a user sensitive preference being successfully guessed becomes

very small, which is equal to $R_0, R_1, R_2, \ldots, R_n$ of the original. In other words, if the attacker cannot find out the user request sequence in advance, it is difficult to guess out the user sensitive preferences (query topics, or query locations).

Table 2. Comparison of existing methods

Method	Accuracy	Efficiency	Usability	Security I	Security II	Security III
Encryption	Good	Good	Not Good	Good	Not Good	Not Good
Pseudonym	Good	Good	Not Good	Good	Not Good	Not Good
Confusion	Not Good	Good	Good	Good	Not Good	Not Good
Dummy	Good	Good	Good	Good	Not Good	Not Good

Based on Observations 1 and 2, we conclude that our method can ensure that the untrusted server cannot analyze and obtain the preference privacy, query privacy and location privacy contained in the submitted request sequences, under the premises of not changing the existing platform architecture, not changing the existing server algorithm, not changing the accuracy of information retrieval, and not changing the efficiency of information retrieval. Finally, based on the six constraints, we make an effectiveness analysis on the four kinds of methods mentioned in the related work section, and the results are shown in Table 2. It can be seen that most of the existing methods are not specifically proposed for personalized information retrieval, thus they cannot meet the actual needs of personalized information retrieval in terms of usability, accuracy, security and efficiency. Moreover, most of existing methods are generally only targeted for a single kind of user privacy, making them not able to meet the Level II Security. In addition, most of existing methods only aim at the user current requests, without considering the historical request data, thus cannot meet the Level III security constraint of personalized information retrieval.

CONCLUSIONS

Now, personalized information retrieval is considered to be an effective tool to solve the problem of information overload and resource addiction. However, along with the rapid development of cloud computing and other emerging network technologies, the problem of user privacy has become a major obstacle to the further developments and applications of personalized information retrieval. In this paper, we propose a basic framework for the privacy protection in personalized information retrieval. The main theoretical contributions are as follows. (1) A unified framework for the privacy protection of personalized information retrieval, which has good practical usability. (2) A privacy model to formulate the constraints that should be satisfied for the effective protection of preference privacy, location privacy and query privacy. (3) An implementation algorithm for the privacy model under the framework, which can comprehensively improve the security of all kinds of user privacy on the untrusted server.

In addition, both theoretical analysis and experimental evaluation demonstrate the effectiveness of our proposed framework, i.e., the practical implication and advantage of our study is that compared with other existing works, it can comprehensively improve the security of all kinds of user privacy on the untrusted server, under the preconditions of not changing the existing platform architecture, and not changing the accuracy and efficiency of personalized information retrieval. Therefore, this paper presents an important study attempt to the protection of user privacy in personalized information retrieval, which is of a positive influence on the problem of privacy and security in the field of organizational and end user computing.

However, there are still some limitations of our work needed to be considered and addressed in future work. (1) The first consideration is the expansion of the basic framework from personalized information retrieval to a wider field of personalized information service. (2) The second consideration is the applicability of the basic framework in a real application environment such as Mobile Terminal. (3) The another consideration is to rebuild and redesign the architecture of personalized information retrieval (including the server), so as to provide a more reliable privacy-preserving personalized retrieval platform.

Funding

The work is supported by the key project of Humanities and Social Sciences in Colleges and Universities of Zhejiang Province (No 2021GH017), Humanities and Social Sciences Project of the Ministry of Education of China (Nos 21YJA870011 and 21YJCZH096), Zhejiang Philosophy and Social Science Planning Project (No 22ZJQN45YB) and National Social Science Foundation of China (No 21FTQB019)

REFERENCES

Arain, Q., Deng, Z., & Memon, I. (2017). Privacy preserving dynamic pseudonym-based multiple mix-zones authentication protocol over road networks. *Wireless Personal Communications*, *95*(2), 1–17. doi:10.1007/s11277-016-3906-4

Arampatzis, A., Drosatos, G., & Efraimidis, P. (2016). Versatile query scrambling for private web search. *Information Retrieval Journal*, *18*(4), 331–358. doi:10.1007/s10791-015-9256-0

Baumeler, A., & Brodbent, A. (2014). Quantum private information retrieval has linear communication complexity. *Journal of Cryptology*, *28*(1), 161–175. doi:10.1007/s00145-014-9180-2

Chatzikokolakis, K., Palamidesi, C., & Stronati, M. (2015). Constructing elastic distinguishability metrics for location privacy. *Proceedings of Privacy Enhancing Technologies Symposium.* doi:10.1515/popets-2015-0023

Dewri, R., & Thurimella, R. (2016). Mobile local search with noisy locations. *Pervasive and Mobile Computing*, *32*, 78–92. doi:10.1016/j.pmcj.2016.04.014

Dong, S. Z., Yang, L., Ding, B., Wu, C. H., & Shao, X. F. (2020). Pricing strategy with customers' privacy concerns in Smart-X systems. *Enterprise Information Systems*, 1–27. doi:10.1080/17517575.2020.1802515

Hewitt, B., & White, G. (2021). Factors influencing security incidents on personal computing devices. *Journal of Organizational and End User Computing*, *33*(4), 185–208. doi:10.4018/JOEUC.20210701.oa9

Li, H., Zhu, Y., Wang, J., Liu, J., Shen, S., Gao, H., & Sun, Y. (2017). Consensus of nonlinear second-order multi-agent systems with mixed time-delays and intermittent communications. *Neurocomputing*, *251*, 115–126. doi:10.1016/j.neucom.2017.04.009

Li, Q., Cao, Z., Ding, W., & Li, Q. (2019). A multi-objective adaptive evolutionary algorithm to extract communities in networks. *Swarm and Evolutionary Computation*, *52*, 100629. doi:10.1016/j.swevo.2019.100629

Li, Q., Cao, Z., Zhong, J., & Li, Q. (2019). Graph representation learning with encoding edges. *Neurocomputing*, *361*, 29–39. doi:10.1016/j.neucom.2019.07.076

Lin, W., Xu, M., He, J., & Zhang, W. (2021). Privacy, security and resilience in mobile healthcare applications. *Enterprise Information Systems*, 1–15. doi:10.1080/17517575.2021.1939896

Liu, J., Wang, X., Shen, S., Yue, G., Yu, S., & Li, M. (2021). A bayesian Q-learning game for dependable task offloading against DDoS attacks in sensor edge cloud. *IEEE Internet of Things Journal*, *8*(9), 7546–7561. doi:10.1109/JIOT.2020.3038554

Liu, J., Wang, X., Yue, G., & Shen, S. (2018). Data sharing in VANETs based on evolutionary fuzzy game. *Future Generation Computer Systems*, *81*, 141–155. doi:10.1016/j.future.2017.10.037

Liu, J., Yu, Y., & Shen, S. (2018). Energy-efficient two-layer cooperative defense scheme to secure sensor-clouds. *IEEE Transactions on Information Forensics and Security*, *13*(2), 408–420. doi:10.1109/TIFS.2017.2756344

Mei, Z., Zhu, H., Cui, Z., Wu, Z., Peng, G., Wu, B., & Zhang, C. (2018). Executing multi-dimensional range query efficiently and flexibly over outsourced ciphertexts in the cloud. *Information Sciences*, *432*, 79–96. doi:10.1016/j.ins.2017.11.065

Meng, W., Lee, B., & Xing, X. (2016). Trackmeornot: Enabling flexible control on web tracking. *Proceedings of International World Wide Web Conference.* doi:10.1145/2872427.2883034

Niu, B., Li, B., & Zhu, Q. (2014). Achieving k-anonymity in privacy-aware location-based services. *Proceedings of IEEE International Conference on Computer Communications.*

Pang, H., Xiao, X., & Shen, J. (2012). Obfuscating the topical intention in enterprise text search. *Proceedings of IEEE International Conference on Data Engineering.* doi:10.1109/ICDE.2012.43

Ravi, N., Krishna, C., & Koren, I. (2019). Enhancing vehicular anonymity in ITS: A new scheme for mix zones and their placement. *IEEE Transactions on Vehicular Technology*, *68*(11), 10372–10381. doi:10.1109/TVT.2019.2936529

Ruchika, G., & Rao, U. (2017). An exploration to location-based service and its privacy preserving techniques: A survey. *Wireless Personal Communications, 96*, 1973-2007.

Saura, J. R., Ribeiro-Soriano, D., & Palacios-Marqués, D. (2021). Evaluating security and privacy issues of social networks based information systems in Industry 4.0. *Enterprise Information Systems*, 1–17. doi:10.1080/17517575.2021.1913765

Shen, S., Huang, L., Zhou, H., Yu, S., Fan, E., & Cao, Q. (2018). Multistage signaling game-based optimal detection strategies for suppressing malware diffusion in fog-cloud-based IoT networks. *IEEE Internet of Things Journal*, *5*(2), 1043–1054. doi:10.1109/JIOT.2018.2795549

Shen, S., Zhou, H., Feng, S., Huang, H., Liu, J., Yu, S., & Cao, Q. (2019). HSIRD: A model for characterizing dynamics of malware diffusion in heterogeneous WSNs. *Journal of Network and Computer Applications*, *146*, 102420. doi:10.1016/j.jnca.2019.102420

Shuai, M., Yu, N., Wang, H., Xiong, L., & Li, Y. (2021). A lightweight three-factor anonymous authentication scheme with privacy protection for personalized healthcare applications. *Journal of Organizational and End User Computing*, *33*(3), 1–18. doi:10.4018/JOEUC.20210501.oa1

Soma, S., Hashem, T., Cheema, M., & Samrose, S. (2017). Trip planning queries with location privacy in spatial databases. *World Wide Web (Bussum)*, *20*(2), 205–236. doi:10.1007/s11280-016-0384-2

Such, J., & Natalia, C. (2018). Multiparty privacy in social media. *Communications of the ACM*, *61*(8), 74–81. doi:10.1145/3208039

Wang, R., Wu, Z., Lou, J., & Jiang, Y. (2021). Attention-based dynamic user modeling and deep collaborative filtering recommendation. *Expert Systems with Applications*, *188*, 116036. doi:10.1016/j.eswa.2021.116036

Wang, S., Hu, Q., Sun, Y., & Huang, J. (2018). Privacy preservation in location-based services. *IEEE Communications Magazine*, *56*(3), 134–140. doi:10.1109/MCOM.2018.1700288

Wang, S., Qin, H., & Sun, Y. (2018). Privacy preservation in location-based services. *IEEE Communications Magazine*, *56*(3), 134–140. doi:10.1109/MCOM.2018.1700288

Wang, T., Bhuiyan, M., Wang, G., Qi, Q., Wu, J., & Hayajneh, T. (2019). Preserving balance between privacy and data integrity in edge-assisted Internet of Things. *IEEE Internet of Things Journal*, *7*(4), 2679–2689. doi:10.1109/JIOT.2019.2951687

Wu, Z., Shen, Lu, C., Li, H., & Su, X. (2021e). How to protect reader lending privacy under a cloud environment: A technical method. *Library Hi Tech*, *39*(2), 1–14.

Wu, Z., Li, G., Liu, Q., Xu, G., & Chen, E. (2018b). Covering the sensitive subjects to protect personal privacy in personalized recommendation. *IEEE Transactions on Services Computing*, *11*(3), 493–506. doi:10.1109/TSC.2016.2575825

Wu, Z., Li, G., Shen, S., Cui, Z., Lian, X., & Chen, E. (2021b). Constructing dummy query sequences to protect location privacy and query privacy in location-based services. *World Wide Web (Bussum)*, *24*(1), 24–45. doi:10.1007/s11280-020-00830-x

Wu, Z., Li, R., Xie, J., Zhou, Z., & Su, X. (2020c). A user sensitive subject protection approach for book search service. *Journal of the Association for Information Science and Technology*, *71*(2), 183–195. doi:10.1002/asi.24227

Wu, Z., Lu, C., Zhao, Y., Xie, J., Zou, D., & Su, X. (2021c). The protection of user preference privacy in personalized information retrieval: Challenges and overviews. *Libri*, *71*(3), 227–237. doi:10.1515/libri-2019-0140

Wu, Z., Shen, S., Li, H., Zhou, H., & Zou, D. (2021d). A comprehensive study to the protection of digital library readers' privacy under an untrusted network environment. *Library Hi Tech*. Advance online publication. doi:10.1108/LHT-07-2021-0239

Wu, Z., Shen, S., Lian, X., Su, X., & Chen, E. (2020a). A dummy-based user privacy protection approach for text information retrieval. *Knowledge-Based Systems*, *195*, 105679. doi:10.1016/j.knosys.2020.105679

Wu, Z., Shi, J., Lu, C., Chen, E., Xu, G., Li, G., Xie, S., & Yu, P. S. (2015). Constructing plausible innocuous pseudo queries to protect user query intention. *Information Sciences*, *325*, 215–222. doi:10.1016/j.ins.2015.07.010

Wu, Z., Wang, R., Li, Q., Lian, X., Xu, G., Chen, E., & Liu, X. (2020b). A location privacy-preserving system based on query range cover-up for loca-tion-based services. *IEEE Transactions on Vehicular Technology*, *69*(5), 5244–5254. doi:10.1109/TVT.2020.2981633

Wu, Z., Xie, J., Lian, X., & Pan, J. (2019a). A privacy protection approach for XML based archives management in a cloud environment. *The Electronic Library*, *37*(6), 970–983. doi:10.1108/EL-05-2019-0127

Wu, Z., Xie, J., Pan, J., & Su, X. (2019b). An effective approach for the protection of user privacy in a digital library. *Libri*, *69*(4), 315–324. doi:10.1515/libri-2018-0148

Wu, Z., Xu, G., Lu, C., Chen, E., Jiang, F., & Li, G. (2018a). An effective approach for the protection of privacy text data in the CloudDB. *World Wide Web (Bussum)*, *21*(4), 915–938. doi:10.1007/s11280-017-0491-8

Wu, Z., Zou, D., Shen, S., & Xu, G. (2021a). An effective approach for the protection of user commodity viewing privacy in e-commerce website. *Knowledge-Based Systems*, *220*, 106952. doi:10.1016/j.knosys.2021.106952

Xue, D., Wu, L., Li, H., Hong, Z., & Zhou, Z.-J. (2017). A novel destination prediction attack and corresponding location privacy protection method in geo-social networks. *International Journal of Distributed Sensor Networks*, *13*(1), 1–16. doi:10.1177/1550147716685421

Zeng, S., Mu, Y., He, M., & Chen, Y. (2018). New approach for privacy-aware location-based service communications. *Wireless Personal Communications*, *11*(2), 1057–1073. doi:10.1007/s11277-018-5748-8

Zhang, H., Shen, S., Cao, Q., Wu, X., & Liu, S. (2020). Modeling and analyzing malware diffusion in wireless sensor networks based on cellular automaton. *International Journal of Distributed Sensor Networks*, *16*(11), 1–9. doi:10.1177/1550147720972944

Zhang, W., Lin, D., Xiao, S., Wu, J., & Zhou, S. (2016). Privacy preserving ranked multi-keyword search for multiple data owners in cloud computing. *IEEE Transactions on Computers*, *65*(5), 1566–1577. doi:10.1109/TC.2015.2448099

Zhou, H., Shen, S., & Liu, J. (2020). Malware propagation model in wireless sensor networks under attack-defense confrontation. *Computer Communications*, *162*, 51–58. doi:10.1016/j.comcom.2020.08.009

Ziegeldorf, J., Henze, M., & Bavendiek, J. (2017). TraceMixer: Privacy-preserving crowd-sensing sans trusted third party. *Proceedings of IEEE Annual Conference on Wireless On-demand Network Systems and Services.* doi:10.1109/WONS.2017.7888771

Zongda Wu is a full professor in Computer Science at Shaoxing University. He received his Ph.D. degree in Computer Science from Huazhong University of Science and Technology (HUST) in 2009. From 2012 to 2014, he worked as a postdoctoral research fellow with the School of Computer Science and Technology at University of Science and Technology of China (USTC). His research interests are primarily in the area of information retrieval and information management.

Shigen Shen received the B.S. degree in fundamental mathematics from Zhejiang Normal University, Jinhua, China, in 1995, the M.S. degree in computer science and technology from Zhejiang University, Hangzhou, China, in 2005, and the Ph.D. degree in pattern recognition and intelligent systems from Donghua University, Shanghai, China, in 2013. He is currently a Professor with the Department of Computer Science and Engineering, Shaoxing University, Shaoxing, China. He is serving as a member of the editorial review board of Journal of Organizational and End User Computing. His current research interests include Internet of Things, cyber security, cloud computing, and game theory.

Huxiong Li is a full professor in Computer Science at Shaoxing University. His current research interests include network security and cloud computing.

Haiping Zhou is a full professor in Computer Science at Shaoxing University. His current research interests include network security and cloud computing.

Chenglang Lu is an associate professor in Computer Science at Zhejiang Institute of Mechanical and Electrical Engineering. His current research interests include network security and cloud computing. He is the corresponding author of this paper.