

Article

A Privacy-Preserving Trajectory Publishing Method Based on Multi-Dimensional Sub-Trajectory Similarities

Hua Shen, Yu Wang and Mingwu Zhang *

School of Computer Science, Hubei University of Technology, Wuhan 430068, China; cshshen@hbust.edu.cn (H.S.); 102111071@hbust.edu.cn (Y.W.)

* Correspondence: csmwzhang@gmail.com

Abstract: With the popularity of location services and the widespread use of trajectory data, trajectory privacy protection has become a popular research area. k -anonymity technology is a common method for achieving privacy-preserved trajectory publishing. When constructing virtual trajectories, most existing trajectory k -anonymity methods just consider point similarity, which results in a large dummy trajectory space. Suppose there are n similar point sets, each consisting of m points. The size of the space is then m^n . Furthermore, to choose suitable $k - 1$ dummy trajectories for a given real trajectory, these methods need to evaluate the similarity between each trajectory in the space and the real trajectory, leading to a large performance overhead. To address these challenges, this paper proposes a k -anonymity trajectory privacy protection method based on the similarity of sub-trajectories. This method not only considers the multidimensional similarity of points, but also synthetically considers the area between the historic sub-trajectories and the real sub-trajectories to more fully describe the similarity between sub-trajectories. By quantifying the area enclosed by sub-trajectories, we can more accurately capture the spatial relationship between trajectories. Finally, our approach generates $k - 1$ dummy trajectories that are indistinguishable from real trajectories, effectively achieving k -anonymity for a given trajectory. Furthermore, our proposed method utilizes real historic sub-trajectories to generate dummy trajectories, making them more authentic and providing better privacy protection for real trajectories. In comparison to other frequently employed trajectory privacy protection methods, our method has a better privacy protection effect, higher data quality, and better performance.



Citation: Shen, H.; Wang, Y.; Zhang, M. A Privacy-Preserving Trajectory Publishing Method Based on Multi-Dimensional Sub-Trajectory Similarities. *Sensors* **2023**, *23*, 9652. <https://doi.org/10.3390/s23249652>

Academic Editor: Hai Dong

Received: 27 October 2023

Revised: 21 November 2023

Accepted: 4 December 2023

Published: 6 December 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: trajectory publishing; privacy preservation; trajectory privacy; k -anonymity

1. Introduction

In the contemporary mobile internet era, individuals increasingly depend on mobile devices and applications for activities such as information access, social networking, and online shopping, all facilitated by the collection and utilization of location data. Furthermore, the advancements in big data, artificial intelligence, and related technologies enable the utilization of users' location data for purposes like optimization of traffic flows [1] and advertising [2], as well as intelligent medical systems [3]. However, collecting location data also raises many security and privacy concerns. Unauthorized use and sharing of location data can result in privacy breaches [4–6]. For instance, individuals may exploit location data for tracking and monitoring purposes [7], including socializing [8], navigation, and travel [9]. Additionally, location data can help rescuers determine affected people's location and movement trajectories in disaster rescue [10], making location data security crucial for public safety. Therefore, protecting personal location privacy has become an essential part of personal privacy protection.

Despite the emergence of numerous location privacy protection techniques, the effectiveness of current approaches faces challenges regarding the relentless technical advancements of attackers [11]. Consequently, researchers have been continuously exploring how to protect users' location privacy more effectively.

Trajectory privacy protection techniques have gained significant attention in research due to their capacity to safeguard the privacy of users' location and behavioral trajectories, becoming an integral component of location privacy protection strategies. There are three fundamental approaches to achieving trajectory privacy: (1) Fake Trajectory [4,12]. This method involves the addition of fabricated trajectory points [12] or the exchange of two location pairs that are close in time and space [4] to obfuscate the authentic trajectory data to preserve privacy. However, this method is susceptible to exploitation by attackers who can discern real from fake trajectory points by scrutinizing time and speed parameters. (2) Sensitive Location Suppression [13]. This approach safeguards user privacy by eliminating or concealing sensitive locations, such as home addresses and workplaces. Nevertheless, if an attacker can access additional information, such as the user's frequently visited locations, the user's actual location can still be deduced. (3) Trajectory k -Anonymity [7,14]. This approach is an extension of k -anonymity, in which the user's trajectory is divided into multiple sub-trajectories, and $k - 1$ other users' trajectories are chosen for each sub-trajectory to obscure the current user's trajectory.

During the process of k -anonymizing trajectories, researchers leverage the similarity of certain trajectory data to anonymize them effectively, preventing attackers from discerning the user's real trajectory. The method described in [15] for publishing an anonymous trajectory is based on the generation of secure starting and ending points, the generation of candidate sets containing secure starting and ending points based on user habits, the generation of $k - 1$ anonymous trajectories in both directions, and the correction of the accessibility of location points on each anonymous trajectory using access probabilities. Zhou and Wang [16] combined a fog computing and k -anonymity approach to protect trajectories, using fog computing to provide users with local storage and mobility and k -anonymity techniques to construct hidden regions for each trajectory snapshot based on time-dependent query probabilities and transfer probabilities to solve the problems of real-time trajectory privacy protection and offline trajectory data protection for continuous queries. Nevertheless, the trajectory k -anonymity technique possesses certain limitations. For instance, attackers can deduce the actual trajectories of users by discerning information disparities (e.g., time, space) among different users [17,18]. As shown in Figure 1, the attacker must distinguish between the user's real trajectory (represented by a solid line) and the first fake trajectory (represented by a dotted line), which the attacker can easily distinguish because the two trajectories take different paths in the middle segment, which is the area through which the attacker distinguishes the user's real trajectory. The attacker cannot distinguish between the real trajectory and the second fake trajectory (represented by a dashed line) because their routes are identical. However, because the real trajectory is generated between 9:00 and 10:00 and the second fake trajectory is generated between 12:00 and 13:00, the attacker can infer the real trajectory of the user using the time information of the trajectories. Thus, it can be seen that the method of selecting trajectories that meet the requirements of trajectory anonymity is the key to trajectory k -anonymity.

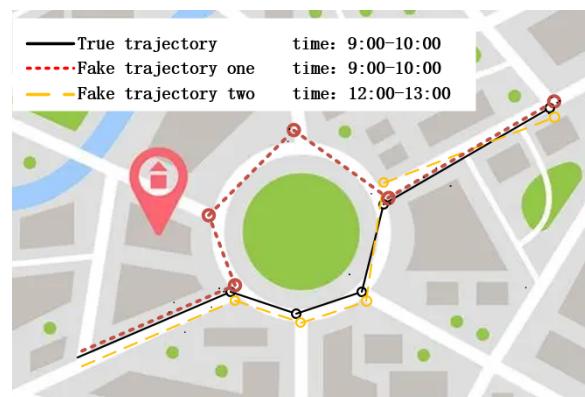


Figure 1. Example of dummy trajectory identification.

To address the shortcomings of trajectory k -anonymity, this study [19] aims to address the privacy leakage problem caused by the increasing amount of personal information sources with the popularization of mobile devices. To prevent the disclosure of user privacy in the published spatio-temporal trajectory dataset, proposing a trajectory k -anonymization model called KPDP, which is based on point density and partitioning. The model optimizes current anonymization methods in regards to trajectory collection, partitioning, preprocessing and trajectory clustering algorithms. It effectively prevents re-identification attacks and minimizes data utility loss in k -anonymized datasets. Zhang [20] proposed the DLIP algorithm, which constructs a dummy location set by randomly selecting probabilistic similarity offset locations under guaranteed semantic distinction, improved security of location privacy, avoiding background knowledge attacks, edge information attacks, and homogeneity attacks to some extent. Yang [21] generated similar trajectories in real-time by the angle and distance between real trajectory points during continuous query service, which can also prevent edge information attacks when protecting user trajectories. Guo [22] introduced a new query privacy algorithm for continuous querying of location-based services that consider users with similar directions, speeds, and the same transmission patterns for anonymization, thus protecting users' privacy throughout the query cycle. The literature [23] introduces a method measuring semantic trajectory similarity across dimensions like weighted time and space. In contrast to conventional methods, it offers a comprehensive view of semantic relationships. Existing trajectory similarity approaches focus on two-dimensional raw trajectories, while literature proposed MSM considers and weights similarity across all dimensions. In summary, we can resist the aforementioned attack by taking into account the similarity of multiple perspectives (such as distance similarity, direction similarity, speed similarity, time similarity, and transmission similarity) between the produced $k - 1$ trajectories and the real trajectory while producing the $k - 1$ trajectories. However, we discover that the majority of the methods discussed above determine how similar two trajectories are by calculating the similarity between position points extracted from the two trajectories. The process of producing $k - 1$ dummy trajectories for a given trajectory usually includes four steps: The first one is to extract the start point, end point, and some other points from the given trajectory. The second one is to generate a point set of size k for each extracted point in which every point is similar to the corresponding extracted point in multiple respects. The third is to produce a dummy trajectory by randomly choosing a point from the point set corresponding to the start point of the given trajectory, the point sets corresponding to the other extracted points, and the point set corresponding to the end point of the given trajectory in turn. The fourth is to determine $k - 1$ dummy trajectories that are most similar to the given trajectory from the produced dummy trajectories. Suppose we extract n points from the given trajectory. We will have n point sets of size k . So, we will generate k^n dummy trajectories. Then, we need to choose $k - 1$ trajectories that are most similar to the given trajectory from the k^n dummy trajectories, which typically results in significant computational overhead.

To address the limitations of trajectory k -anonymity with minimal computational overhead, this paper introduces a privacy-preserving trajectory publishing method (PP-TPS) that relies on multi-dimensional similarities among sub-trajectories. PP-TPS is a method for achieving trajectory k -anonymity. To minimize the computational burden associated with creating $k - 1$ similar trajectories from a given trajectory, leveraging historic trajectory data, PP-TPS initially divides both the target trajectory and the historic data into sub-trajectories. Then, PP-TPS filters historic sub-trajectories for each sub-trajectory of the trajectory according to the computed multi-dimensional similarities between them. The main contributions of this paper are as follows:

- We design a k -anonymous privacy-preserving method (PP-TPS) based on sub-trajectories' similarity. PP-TPS divides a given real trajectory in chronological order to obtain multiple segments (that is, sub-trajectories). Then, PP-TPS finds all sub-trajectories that are similar to these segments from historic trajectory sets. Finally, PP-TPS generates dummy trajectories for the given real trajectory by splicing similar sub-trajectories in

chronological order. In other words, PP-TPS generates dummy trajectories by using real historic sub-trajectories to improve the authenticity of the generated dummy trajectories, enhancing privacy protection for real trajectories.

- PP-TPS considers the similarity of points between the historic trajectory and the real trajectory but also considers the similarity of the sub-trajectories between the historic and real trajectories. Moreover, PP-TPS proposes to measure the similarity of sub-trajectories by the area distance between them. This processing skill makes the processing process simpler and more efficient by drastically reducing the dummy trajectory space formed by simply taking the point similarity into account and skillfully avoiding solving the entire trajectory similarity.
- PP-TPS present a novel multidimensional similarity calculation approach that surpasses conventional similarity measures. It not only considers trajectory similarities in space but also factors in multiple dimensions. This cutting-edge method allows for a more comprehensive and precise depiction of trajectory relationships, offering a detailed and profound perspective for trajectory research.
- PP-TPS conducts comprehensive experiments on real datasets, and the results show that our method is more effective in terms of data availability and privacy protection compared to other methods.

The structure of the paper is outlined as follows: Section 2 provides an overview of related work. Section 3 introduces the relevant concepts and notations employed throughout the paper. Section 4 offers a comprehensive description of the proposed method, PP-TPS. Section 5 presents a comparative analysis of PP-TPS's performance against other methods. Lastly, Section 6 summarizes the paper.

2. Related Work

The concept and basic principles of k -anonymity were first proposed by Latanya Sweeney [24]. k -anonymity technology aims to protect personal privacy by preventing an attacker from uniquely identifying a specific user. Gruteser and Grunwald [25] first applied k -anonymity methods to location services. They proposed a method to ensure users' real locations cannot be accurately determined by temporal masking of the location information in time and space. Subsequently, k -anonymity techniques have gained significant traction in the domain of location-based services, with applications ranging from privacy-preserving traffic monitoring and management in traffic systems using k -anonymity technologies [26] to preventing the leakage of node location information in sensor networks through the adoption of k -anonymity technologies [27]. These techniques have also been instrumental in protecting user search history within search engines [28], securing the trajectories of devices [29], and preserving location privacy in mobile social networks [30]. This paper focuses on trajectory privacy by utilizing k -anonymization technologies. In summary, k -anonymization methods for trajectory privacy can be divided into two types: discrete point-based k -anonymization methods and trajectory-based k -anonymization methods.

2.1. Discrete Point-Based k -Anonymization

The discrete point-based approach [31–33] typically involves discretizing a trajectory into a sequence of points and then perturbing each point to attain k -anonymity, thereby enabling the creation of k -anonymized trajectories. The discrete point-based approaches are simple and intuitive. According to the dimension of similarity, this approach can be divided into two categories: discrete point-based k -anonymization with one-dimensional similarity and discrete point-based k -anonymization with multi-dimensional similarity.

The one-dimensional similarity k -anonymization method considers the similarity of locations from one perspective, such as spatial, semantic, or probabilistic, simplifying implementation. For example, the paper [34] proposed a new method called ARB to protect location privacy. The ARB method divides the space into grids and calculates the probability of queries within different grids based on historical data. Then, the query probabilities are used to generate anonymous regions for users to protect their location privacy. In [35],

a location k -anonymity privacy protection scheme for vehicular networks was introduced. It involves dividing locations into multiple cells and assigning each cell a query probability based on the historical query distribution. By utilizing the Fréchet distance to measure trajectory similarity, it identifies the top $4k$ locations with the highest probabilistic similarity as candidates and delivers the optimal set of k -anonymous locations for straightforward and effective location privacy protection. In [29], the authors derived a gravity model describing user movement patterns from stored trajectory data. They established the flow relationships between locations, computed transfer probabilities among different grids based on the derived gravity model, and identified the virtual location with the highest location entropy to generate $k - 1$ virtual locations, thereby preserving user trajectory data. Ref. [31] utilized a random walk to generate virtual location points within a specific geographic distance to ensure that all users in the steganography region have the same probability of being identified as real users. The scheme provided by [32] considered the user's sensitivity to different location semantics and optimized collaborative segmentation to protect location privacy.

Although location points k -anonymity with similarity of one-dimensional location metric is easy to implement, an attacker can still threaten the user's privacy with some side information. For example, an attacker may find that a user posts a post about a restaurant located in a certain area at one point in time, followed by a post about a movie theater located in a certain area at another point in time. By analyzing the temporal and geographical connections between these posts, the attacker can deduce the user's unique trajectory based on the temporal references and location semantics (restaurant, movie theater), even when the user's location has been anonymized. Researchers filtered them for multi-dimensional similarity to get virtual locations that are more similar to real locations. Literature [36] presents Enhanced-DLP, a novel, lightweight false location privacy protection scheme, aimed at tackling computational cost and additional information attacks challenges found in traditional schemes. Enhanced-DLP efficiently selects false locations to create k -anonymous sets using an improved greedy algorithm. The method incurs lower computational costs in selecting false locations and is capable of defending against attacks using supplementary information, in contrast with current approaches. The paper [37] considered the side information that the adversary may exploit. It first selected these virtual locations based on the entropy metric and subsequently introduced an improved DLS algorithm to maximize the dispersion of these selected virtual locations. Based on this, Yang et al. [38] took query probability and semantic location information as critical parameters, presented a virtual location selection model to evaluate the quality of virtual locations, and used a genetic algorithm-based optimization method to find the optimal solution, which ends up with a set of virtual locations with query probability as close as possible and also makes the locations in them as semantically and physically dispersed as possible. Unlike the above, the paper [39] built a semantic location tree (LST) and converted the semantic distance into the number of hops between nodes in the LST, designing a method that takes into account the semantic diversity and physical dispersion of locations, and combined the two objectives of geographic location and location semantics into a single objective optimization problem to improve efficiency, and ultimately produced an anonymous region. Concerning multi-dimensional location point similarity, Ref. [40] proposed a semantic-based method to protect trajectory anonymity. The method constructed a semantic region of sensitive location points by modeling each location point in the trajectory, where each sensitive point is associated with $k - 1$ similar POI points. The method compared spatiotemporal and semantic similarities for anonymization to generate a usable anonymized trajectory dataset.

To select the $k - 1$ dummy trajectories closest to the supplied trajectory, this approach must consider various virtual trajectories created by orderly combining location points of candidate point sets (anonymous point sets) corresponding to the selected locations on the given trajectory. This process often leads to considerable computational overhead.

2.2. Trajectory-Based k -Anonymization

The discrete-point k -anonymity-based approach significantly impacts trajectory accuracy and entails higher computational complexity, necessitating the processing of all individual points on the trajectory. In contrast, the trajectory k -anonymity-based approach treats a sequence of discrete points within the trajectory holistically to achieve k -anonymity, focusing on the trajectory's overall characteristics for enhanced user privacy protection. This approach efficiently reduces computational overhead and enhances user privacy protection. Paper [41] presents a novel distributed k -anonymity algorithm to address the problem of location privacy leaks. The algorithm evaluates the similarity of users by analyzing their points of interest and social conduct, and picks users with a high similarity to form a group of anonymous collaborators. Ultimately, a homogenization algorithm guarantees the relative uniform distribution of the anonymized location points. Unlike the above scheme, ref. [42] primarily constrained the geographic location of trajectories and introduced a privacy-preserving method for measuring distance-based trajectory similarity. This method entailed constructing a trajectory network with spatial similarity serving as weights and selecting the node with the highest weight as the clustering center to create clusters meeting specific conditions. In the process of trajectory point perturbation, the trajectory points are shifted to the positions of neighboring nodes according to their degrees and specific conditions to achieve anonymization. This research [43] introduces sequential triple decision making and dynamic k -values to traditional k -anonymity to achieve personalized privacy protection. The proposed multi-level personalized k -anonymization model constructs a hierarchical decision table with attribute generalization trees and sensitive decision values supplied by the user. It utilizes dynamic k -value sequences to anonymize data. This study presents an extensive model for safeguarding privacy across multiple levels, thereby augmenting the use of sequential triple decision making in the privacy protection domain. Ref. [44] fused the similarity of multiple dimensions into trajectory similarity and proposed a personalized anonymization model that considers trajectory privacy and data utility. The model considers the effects of trajectory similarity, direction, and the trajectory distance on privacy and data utility and achieves a near-optimal set of k -anonymized trajectories by transforming the k -optimal trajectory selection problem into a constrained minimum spanning tree problem and using a trajectory graph model with a greedy strategy. In addition, the model pays full attention to users' characteristics and requirements, exhibits strong scalability, and proves to be well suited to practical application scenarios. To further enhance the privacy protection effect of trajectories, this research [45] proposes the Privacy-Enhanced Distributed k -Anonymous Reward Mechanism (PEAK) to incentivize users to participate in distributed k -anonymity privacy protection in location services. PEAK does not need to trust third parties, establishes anonymous zones through currency transactions and location transmission, introduces role recognition and accountability mechanisms to limit malicious users, improves security and feasibility, and successfully establishes anonymous zones with a success rate of over 0.9, significantly reducing the utility of malicious users. Traditional k -anonymous trajectory privacy-preserving methods based on trajectory similarity usually use only one or two similarities between trajectories, resulting in ignoring the relationship between other attributes in the trajectories, and thus they cannot achieve the expected privacy-preserving effect.

For the above issues, the proposed PP-TPS method generates dummy trajectories by combining candidate sub-trajectories instead of candidate location points to reduce the computation cost and filters historic sub-trajectories according to multi-dimensional similarities (including time, space, semantics, direction, and probability) when creating candidate sub-trajectory sets to improve the intensity of privacy protection of user trajectories.

3. Preliminaries

3.1. Related Concepts

- Trajectory

A trajectory is the path of an object moving through time and space. A trajectory usually consists of a series of coordinate points and a timestamp. A trajectory can be represented formally as $P = \{p_1, \dots, p_n\}$, and suppose we choose n location points of the trajectory to represent it. Here, p_1 and p_n are the trajectory's start and end, respectively. p_2, p_3, \dots, p_{n-1} are local extremum points of the trajectory. Moreover, $p_i = (s_i, L_i, T_i)$, $s_i = (x_i, y_i)$ is the latitude and longitude of the sample location point p_i ; L_i is p_i 's semantic information; and T_i is its timestamp.

- Sub-trajectory

A sub-trajectory is a series of consecutive location points intercepted from a trajectory. In this paper, we refer specifically to a series of two adjacent location points. For example, $st_i = \{p_i, p_{i+1}\}$ is the i th sub-trajectory of the trajectory P , where $i = 1, 2, \dots, n - 1$.

- Local extremum point

A local extremum point is defined as the start point or end point of a sub-trajectory in a trajectory whose direction of motion is different from that of the adjacent sub-trajectories. p_i is a local extremum point if it satisfies the following condition:

$$|k_{(i-1,i)} - k_{(i,i+1)}| > |k_{(i-1,i)}|, \quad (1)$$

where $k_{(i-1,i)}$ denotes the slope from p_{i-1} to p_i .

3.2. Extracting Similar Location Points

PP-TPS filters similar location points out of all location points mainly based on their resemblance in terms of time, space, and semantics. If two points, denoted as p_i and p_j , satisfy the requirements listed below, PP-TPS classifies them as similar.

$$\begin{cases} Sim_{spa}(p_i, p_j) > \delta_{spa} \\ Sim_{time}(p_i, p_j) > \delta_{time} \\ Sim_{sem}(p_i, p_j) > \delta_{sem} \end{cases} \quad (2)$$

where $Sim_{spa}(p_i, p_j)$, $Sim_{time}(p_i, p_j)$, $Sim_{sem}(p_i, p_j)$ are the spatial similarity, the temporal similarity, and the semantic similarity between p_i and p_j , respectively. In addition, δ_{time} , δ_{spa} , and δ_{sem} represent the thresholds related to spatial similarity, temporal similarity, and semantic similarity, respectively. These thresholds influence the selection of location points, ensuring that the filtered points exhibit significant similarities with the actual location points concerning spatial, temporal, and semantic attributes.

Below, we describe how to measure points' spatial similarity, temporal similarity, and semantic similarity.

- Spatial similarity [21]

Spatial similarity $Sim_{spa}(p_i, p_j)$ refers to the degree of geographic closeness between two points. In our method, we calculate spatial similarity by employing the Euclidean distance:

$$Sim_{spa}(p_i, p_j) = 1 - \frac{d(p_i, p_j)}{1 + d(p_i, p_j)} = \frac{1}{1 + d(p_i, p_j)}, \quad (3)$$

where $d(p_i, p_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ is the Euclidean distance between p_i and p_j .

- Temporal similarity

Temporal similarity $Sim_{time}(p_i, p_j)$ refers to the degree of similarity between two points in time. The formula for temporal similarity is:

$$Sim_{time}(p_i, p_j) = \frac{1}{1 + d_{time}(p_i, p_j)}, \quad (4)$$

where $d_{time}(p_i, p_j) = |T_i - T_j|$, T_i (respectively, T_j) is the corresponding timestamp of the point p_i (respectively, p_j).

- Semantic similarity

Semantic similarity $Sim_{sem}(p_i, p_j)$ refers to the similarity between two points in the semantic space. PP-TPS measures the semantic similarity between two points $p_i = \{s_i, L_i, T_i\}$ and $p_j = \{s_j, L_j, T_j\}$ based on WordNet.

WordNet [20,46] is an English vocabulary database containing a large number of English words and the relationships between them and their semantics. The vocabulary in WordNet can be categorized according to semantic classes, each of which contains words with the same or similar meanings. Words in WordNet may contain more than one semantic meaning, so it is necessary to consider multiple semantics when we calculate the location semantic similarity between p_i and p_j .

Location semantics pertain to the semantic categorization of a location point, symbolized as L . Using geo-tags, these location semantics have been categorized into ten separate categories for the sake of this study. These categories include residential areas, commercial activities, leisure and entertainment, education and culture, and transportation. The acquisition of location semantics is contingent upon map information. The detailed information is shown in Table 1 [30].

Table 1. Location semantics categories.

Semantic Category	Detailed Information
Transportation	Airport, railway, bus station
Education	Campus, training center
Work	Government, office, building, company
Home	Apartment, house
Industry	Factory, industrial estate, etc.
Health care	Hospital, medical center, pharmacy
Food	Restaurant
Entertainment	Hotel, gym, leisure
Shopping	Shop, mall, outlet
Outdoor	Park, sports field

If $L_i = L_j$, the semantic distance $Sim_{sem}(p_i, p_j) = 1$; otherwise, $Sim_{sem}(p_i, p_j)$ [20,46] is equal to

$$\frac{\sum_{l_i \in L_i} (\max_{l_j \in L_j} Sim_{sem}(l_i, l_j)) + \sum_{l_j \in L_j} (\max_{l_i \in L_i} Sim_{sem}(l_j, l_i))}{|L_i| + |L_j|} \quad (5)$$

where $|L_i|, |L_j|$ denotes the number of semantic meanings of L_i and L_j , respectively.

3.3. Extracting Similar Sub-Trajectories

PP-TPS uses the area enclosed by a real sub-trajectory and a historic sub-trajectory to measure the similarity of the two sub-trajectories. The size of the area is called the “area distance” between the two sub-trajectories. When the area distance is smaller, PP-TPS

deems them to be closer. If the area distance $\text{Area}(\text{subtr}_i, \text{subtr}_j)$ between the two sub-trajectories subtr_i and subtr_j is less than a given threshold δ_{area} , PP-TPS considers that they are similar sub-trajectories.

$$\text{Area}(\text{subtr}_i, \text{subtr}_j) < \delta_{\text{area}} \quad (6)$$

Note that the start points and end points of the two sub-trajectories are similar points. The area threshold δ_{area} is determined by the system.

We use the triangulation method [47] to evaluate the area enclosed by two sub-trajectories. We give an example (shown in Figure 2) to illustrate the specific evaluation process. Suppose, in Figure 2, the real sub-trajectory is $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, and the historic sub-trajectory is $\{(a_1, b_1), (a_2, b_2), \dots, (a_m, b_m)\}$, where $(x_1, y_1) = (a_1, b_1)$, $(x_n, y_n) = (a_m, b_m)$ and n' and m' are usually small integers. First, we obtain the area of the triangle formed by the three location points (x_i, y_i) , (x_{i+1}, y_{i+1}) , (a_2, b_2) by calculating $S_i = \frac{1}{2} \cdot (x_i y_{i+1} + x_{i+1} b_2 + a_2 y_i - x_i b_2 - x_{i+1} y_i - a_2 y_{i+1})$, where i is between 1 and $n' - 1$. Then, we calculate the area S'_n of the triangle formed by (a_2, b_2) , $(a_{m'-1}, b_{m'-1})$, $(a_{m'}, b_{m'})$, the area $S_{n'+1}$ of the triangle formed by (a_2, b_2) , $(a_{m'-2}, b_{m'-2})$, $(a_{m'-1}, b_{m'-1})$, \dots , and the area $S_{n'+m'-4}$ of the triangle formed by (a_2, b_2) , (a_3, b_3) , (a_4, b_4) . Finally, the area of the two sub-trajectories is $S = S_1 + S_2 + \dots + S_{n'+m'-4}$.

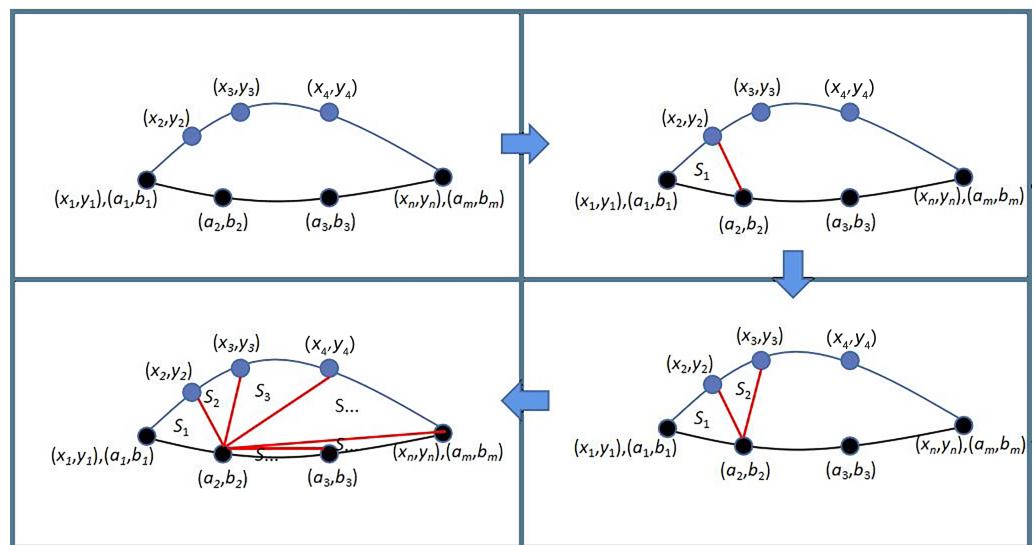


Figure 2. Example of evaluating area size.

4. PP-TPS Description

The main symbols used in this article and their explanations are shown in Table 2.

4.1. Method Framework

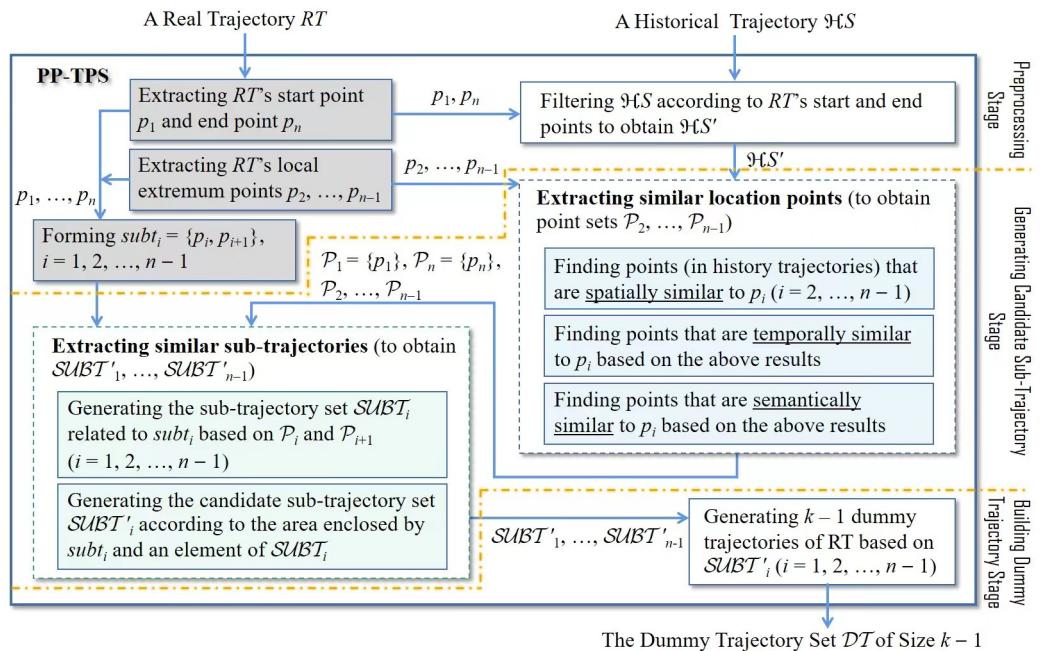
The objective of PP-TPS is to establish k -anonymity for a provided trajectory, denoted as RT , using a given historic trajectory set \mathcal{HS} , prior to the publication of trajectories. Figure 3 illustrates PP-TPS's framework and workflow. PP-TPS consists of three stages: the preprocessing stage, the generating a candidate sub-trajectory stage, and the building a dummy trajectory stage.

In the preprocessing stage, PP-TPS extracts RT 's start point p_1 , end point p_n , and local extremum points p_2, \dots, p_{n-1} . Subsequently, PP-TPS intercepts the part of RT from p_i to p_{i+1} to take as subtr_i and chooses the trajectories that are the same as the start and end points of RT from \mathcal{HS} . The set of filtered trajectories is denoted as \mathcal{HS}' .

Table 2. Main notation.

Symbol	Explanation
$RT = \{p_1, \dots, p_n\}$	A real trajectory
\mathcal{HS}	A historic trajectory set
\mathcal{HS}'	The historic trajectory set obtained after initially screening \mathcal{HS}
\mathcal{H}	A historic trajectory in \mathcal{HS}'
$p_{\mathcal{H}}$	A location point on \mathcal{H}
\mathcal{P}_i	A point set related to p_i ($i = 1, 2, \dots, n$)
$subt_i = \{p_i, p_{i+1}\}$	The sub-trajectory between points p_i and p_{i+1} of RT ($i = 1, 2, \dots, n - 1$)
$SUBT'_i$	The candidate sub-trajectory set filtered out from $SUBT_i$
\mathcal{DT}	The dummy trajectory set of size $k - 1$ for RT
$\delta_{spa}, \delta_{time}$	The spatial threshold and the temporal threshold
$\delta_{sem}, \delta_{area}$	The semantic threshold and the area threshold
$p_{\mathcal{H}}.T$	The time attribute of point $p_{\mathcal{H}}$
$p_i.T$	The time attribute of point p_i
$p_{i,j}$	The j th location point of \mathcal{P}_i

(1) p_1 is RT 's start point, p_n is RT 's end point, and p_i ($i = 2, 3, \dots, n - 1$) is RT 's local extremum point. (2) We suppose that every trajectory of \mathcal{HS} has a unique label. (3) Every trajectory belonging to \mathcal{HS}' has the same start and end points as RT . (4) Every point of \mathcal{P}_i (extracted from \mathcal{HS}') is similar to p_i in three dimensions (such as space, time, semantics). Each node in set \mathcal{P}_i carries a label that is the label of the historical trajectory from which it came. \mathcal{P}_1 (respectively, \mathcal{P}_n) includes only one point p_1 (respectively, p_n) with a special label which means that the point comes from all trajectories in \mathcal{HS}' . (5) $sp_i \in \mathcal{P}_i$, $sp_{i+1} \in \mathcal{P}_{i+1}$, and the label attached to sp_i are the same as sp_{i+1} . Therefore, $\{sp_i, sp_{i+1}\}$ is a part of the historic trajectory ($\in \mathcal{HS}'$) labeled by this label. (6) If the area closed by a sub-trajectory belonging to $SUBT_i$ and $subt_i$ is less than the threshold δ_{area} , then the sub-trajectory is a member of $SUBT'_i$.

**Figure 3.** Overall flowchart of PP-TPS.

After that, PP-TPS enters the “generating a candidate sub-trajectory stage”. In this stage, PP-TPS first finds all points on the trajectories included by \mathcal{HS}' that are similar to p_i ($i = 2, \dots, n - 1$) in the aspects of time, space, and semantics, and uses \mathcal{P}_i to denote the set of the found points. Note that \mathcal{P}_1 and \mathcal{P}_n are set to $\{p_1\}$ and $\{p_n\}$, respectively. Subsequently, PP-TPS searches a point sp_i from \mathcal{P}_i and a point sp_{i+1} from \mathcal{P}_{i+1} ($i = 1, 2, \dots, n - 1$), both of which have the same label, indicating that sp_i and sp_{i+1} come from the same historic trajectory. If PP-TPS finds such sp_i and sp_{i+1} , it intercepts the segment of the corresponding historic trajectory between sp_i and sp_{i+1} and pushes this segment into $SUBT_i$. Then, PP-TPS further filters the sub-trajectories of $SUBT_i$ based on their enclosed areas with $subt_i$ to obtain the candidate sub-trajectory set $SUBT'_i$. Suppose the size of the space contained with one sub-trajectory of $SUBT_i$ and $subt_i$ is less than δ_{area} . In that case, PP-TPS considers that this sub-trajectory is similar to $subt_i$, a candidate used to create RT’s dummy trajectories.

In last stage, PP-TPS creates $k - 1$ dummy trajectories of RT based on candidate sets $SUBT'_1, SUBT'_2, \dots, SUBT'_{n-1}$.

4.2. Preprocessing Stage

This stage mainly involves processing a given real trajectory RT and coarse screening a given historic trajectory \mathcal{HS} . The specific operations of preprocessing include:

- Step 1: PP-TPS extracts the beginning point p_1 and the ending point p_n of RT (suppose PP-TPS will extract a total of n location points from RT).
- Step 2: PP-TPS, according to p_1 and p_n , narrows down the given historic trajectory \mathcal{HS} to obtain a preliminary candidate historic trajectory set \mathcal{HS}' . The beginning and ending of each trajectory in \mathcal{HS}' are p_1 and p_n , respectively.
- Step 3: PP-TPS, according to Equation (1), picks out local extremum points p_2, \dots, p_{n-1} from RT and intercepts the part of RT from p_i to p_{i+1} to take as $subt_i$ ($i = 1, 2, \dots, n - 1$).

4.3. Generating a Candidate Sub-Trajectory Stage

The main task of this stage is to generate the candidate sub-trajectory set $SUBT'_i$ for $subt_i$ ($i = 1, 2, \dots, n - 1$). To achieve this, PP-TPS first carries out the *ALPOINT* algorithm (Algorithm 1) to obtain the point sets \mathcal{P}_i , which include the points extracted from historic trajectories in \mathcal{HS}' and are similar to p_i ($i = 1, \dots, n$) in the aspects of time, space, and semantics. Then, according to \mathcal{P}_i , PP-TPS runs the *ALSUBT* algorithm (Algorithm 2) to generate the candidate sub-trajectory set, $SUBT'_i$.

Algorithm 1 Extracting similar points AL_{POINT} .

Input: \mathcal{HS}' ; p_1, p_2, \dots, p_n ; Δt ; δ_{spa} , δ_{time} , δ_{sem}
Output: $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_n$.

- 1: Let $\mathcal{P}_1 \leftarrow \emptyset, \dots, \mathcal{P}_n \leftarrow \emptyset$;
- 2: Set a certain time offset Δt ;
- 3: **for** $i = 1$ to n **do**
- 4: **for all** $\mathcal{H} \in \mathcal{HS}$ **do**
- 5: **if** $|p_{\mathcal{H}}.T - p_i.T| < \Delta t$ **then**
- 6: $\mathcal{P}_i \leftarrow \mathcal{P}_i \cup \{p_{\mathcal{H}}\}$;
- 7: **end if**
- 8: **end for**
- 9: **end for**
- 10: **for** $i = 1$ to n **do**
- 11: **for all** $p_{i,j} \in \mathcal{P}_i$ **do**
- 12: Calculate $Sim_{spa}(p_i, p_{i,j})$ according to Equation (3);
- 13: **if** $Sim_{spa}(p_i, p_{i,j}) \leq \delta_{spa}$ **then**
- 14: Remove $p_{i,j}$ from \mathcal{P}_i ;
- 15: **end if**
- 16: **end for**
- 17: **end for**
- 18: **for** $i = 1$ to n **do**
- 19: **for all** $p_{i,j} \in \mathcal{P}_i$ **do**
- 20: Calculate $Sim_{time}(p_i, p_{i,j})$ according to Equation (4);
- 21: **if** $Sim_{time}(p_i, p_{i,j}) \leq \delta_{time}$ **then**
- 22: Remove $p_{i,j}$ from \mathcal{P}_i ;
- 23: **end if**
- 24: **end for**
- 25: **end for**
- 26: **for** $i = 1$ to n **do**
- 27: **for all** $p_{i,j} \in \mathcal{P}_i$ **do**
- 28: Calculate $Sim_{sem}(p_i, p_{i,j})$ according to Equation (5);
- 29: **if** $Sim_{sem}(p_i, p_{i,j}) \leq \delta_{sem}$ **then**
- 30: Remove $p_{i,j}$ from \mathcal{P}_i ;
- 31: **end if**
- 32: **end for**
- 33: **end for**
- 34: **return** $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_n$;

Algorithm 2 Extracting a candidate sub-trajectory AL_{SUBT} .

Input: $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_n; \mathcal{HS}'; subt_1, subt_2, \dots, subt_{n-1}$
Output: $SUBT'_1, SUBT'_2, \dots, SUBT'_{n-1}$

```

1: Let  $SUBT'_1 \leftarrow \emptyset, \dots, SUBT'_{n-1} \leftarrow \emptyset;$ 
2: for  $i = 1$  to  $n - 1$  do
3:   for  $j = 1$  to  $|\mathcal{P}_i|$  do
4:     for  $k = 1$  to  $|\mathcal{P}_{i+1}|$  do
5:       if  $\mathcal{P}_i$ 's  $j$ th point and  $\mathcal{P}_{i+1}$ 's  $k$ th point have same trajectory label then
6:         Find the historic trajectory from  $\mathcal{HS}'$ ;
7:         Intercept the segment of the trajectory between the two points;
8:         Add this segment into  $SUBT'_i$ ;
9:       end if
10:      end for
11:    end for
12:  end for
13:  for  $i = 1$  to  $n - 1$  do
14:    for  $j = 1$  to  $|mathcal{SUBT}'_i|$  do
15:      Calculate the area  $S$  enclosed by the  $i$ th item of  $SUBT'_i$  and  $subt_i$  by running the
         $AL_{AREA}$  algorithm (Algorithm 3);
16:      if  $S \geq \delta_{area}$  then
17:        Remove the item from  $SUBT'_i$ ;
18:      end if
19:    end for
20:  end for
21: return  $SUBT'_1, SUBT'_2, \dots, SUBT'_{n-1}$ ;

```

Algorithm 3 Calculating the Area AL_{AREA} .

Input: seg_1, seg_2
Output: The area S enclosed by seg_1 and seg_2 .

```

1: Let  $S \leftarrow 0$ ;
2: Take  $n'$  points  $(p_1^{(1)}, p_2^{(1)}, \dots, p_{n'}^{(1)})$  from  $seg_1$ ;
3: Take  $m'$  points  $(p_1^{(2)}, p_2^{(2)}, \dots, p_{m'}^{(2)})$  from  $seg_2$ ;
4: for  $i = 1$  to  $n' - 1$  do
5:   Calculate the area  $S_{\Delta}$  of the triangle defined by points  $p_2^{(2)}, p_i^{(1)}, p_{i+1}^{(1)}$ ;
6:    $S \leftarrow S + S_{\Delta}$ ;
7: end for
8: for  $i = m' - 1$  to 4 do
9:   Calculate the area  $S_{\Delta}$  of the triangle defined by points  $p_2^{(2)}, p_i^{(2)}, p_{i-1}^{(2)}$ ;
10:   $S \leftarrow S + S_{\Delta}$ ;
11: end for
12: return  $S$ ;

```

4.4. Building a Dummy Trajectory Stage

In this stage, based on the candidate sub-trajectory sets ($SUBT'_1, SUBT'_2, \dots, SUBT'_{n-1}$), PP-TPS can generate $k - 1$ dummy trajectories for the real trajectory RT by adopting various strategies, which, together with the user's real trajectories, form an anonymized set containing k trajectories in order to k -anonymize the real trajectories, such as randomized policy, top- $k - 1$ policy, and so on. Here, we use the randomized policy to create dummy trajectories according to similar historic sub-trajectories. PP-TPS runs the AL_{DUMT} algorithm (Algorithm 4) to realize k -anonymity for RT .

Algorithm 4 Building a dummy trajectory AL_{DUMT} .**Input:** $SUBT'_1, SUBT'_2, \dots, SUBT'_{n-1}; k$ **Output:** The dummy set DT of size $k - 1$.

```

1: Let  $DT \leftarrow \emptyset$ ;
2: for  $i = 1$  to  $k - 1$  do
3:   for  $j = 1$  to  $n - 1$  do
4:     if  $j$  is equal to 1 then
5:       Randomly choose a sub-trajectory  $st_j$  from  $SUBT'_j$ ;
6:     else
7:       Choose a sub-trajectory  $st_j$  from  $SUBT'_j$  and ensure that  $st_j$ 's beginning is
        $st_{j-1}$ 's ending;
8:     end if
9:   end for
10:  Add the dummy trajectory  $\{st_1, st_2, \dots, st_{n-1}\}$  into  $DT$ ;
11: end for
12: return  $DT$ ;

```

4.5. Trajectory Publishing Method PP-TPS

PP-TPS adopts the k -anonymity mechanism to achieve the secure release of a given trajectory RT . To achieve this, PP-TPS introduces an innovative approach to establish trajectory k -anonymity. The AL_{PP-TPS} algorithm (Algorithm 5) illustrates the entire process.

Algorithm 5 Secure trajectory publishing AL_{PP-TPS} .**Input:** RT, HS **Output:** The release set PT .

```

1: Let  $PT \leftarrow \{RT\}$ ;
2: Implement preprocess operations to obtain  $p_1, \dots, p_n; HS', subt_1, \dots, subt_{n-1}$ ;
3: Set the values of  $\Delta t, \delta_{spa}, \delta_{time}, \delta_{sem}, \delta_{area}, k$ ;
4: Run  $AL_{POINT}(p_1, \dots, p_n; HS'; \Delta t, \delta_{spa}, \delta_{time}, \delta_{sem})$  to get  $P_1, P_2, \dots, P_n$ ;
5: Carry out  $AL_{SUBT}(P_1, P_2, \dots, P_n; HS'; subt_1, \dots, subt_{n-1})$  to obtain  $SUBT'_1,$ 
 $SUBT'_2, \dots, SUBT'_{n-1}$ ;
6: Implement  $AL_{DUMT}(SUBT'_1, SUBT'_2, \dots, SUBT'_{n-1}; k)$  to get  $DT$ ;
7:  $PT \leftarrow PT \cap DT$ ;
8: return  $PT$ ;

```

5. Experiment Analysis**5.1. Experimental Database**

The trajectory dataset used in this study was sourced from the Geolife project [48] conducted by Microsoft Research Asia. The data were collected over a period of five years from April 2007 to August 2012 by 182 users. The dataset includes user ID, trajectory dates, latitude, longitude, and the time of each location point in the GeoLife format. It comprises 17,621 trajectories primarily originating from Beijing, China. Our goal is to evaluate the efficiency of PP-TPS in achieving k -anonymized trajectories using this dataset. Figure 4 provides separate visual representations of the trajectory distribution for the 14th user in the Geolife dataset, both on a GPS map (Figure 4a) and within the latitude/longitude coordinate system (Figure 4b).

The GeoLife trajectory dataset does not include location names in the location points of the trajectories. To add this information, we used a process called inverse geocoding. This allows us to assign semantics and attributes to the location points. However, because the time intervals between location points are so short, multiple location points can refer to a single location. To deal with this, we selected the location points with the same location name and those that are closer in time to the midpoint for computation. We then remove the other location points. We filtered a total of 4245 trajectories from the original data, which belonged to 30 users. Location semantics were then integrated into the location

points on these trajectories to facilitate the calculation of semantic similarity. Please refer to Table 3 for statistics regarding the actual dataset used.

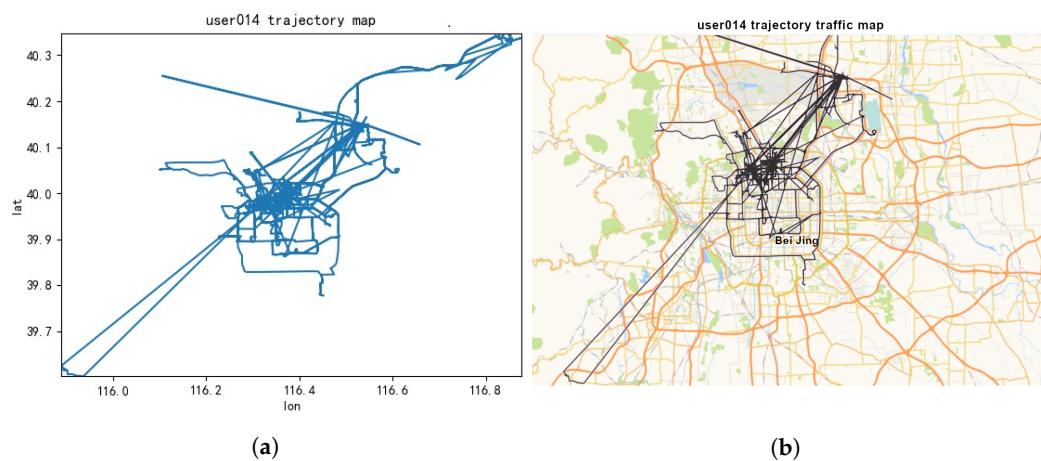


Figure 4. Trajectory map of the 14th user. (a) GPS map; (b) latitude and longitude coordinate system.

Table 3. Geolife statistics data.

Geolife Statistics Data	Value
Number of trajectories	4242
Average trajectory length	1662.61
Number of location points	7,052,782
Average length of time per trajectory (hours)	3.78

Experiments were conducted using the PyCharm 2022 development platform, and the algorithms were implemented in Python. The experiments were conducted on a hardware environment that consisted of an Intel Core i5-13600KF 3.50 GHz processor and a Windows 11 64 GB RAM operating system. The manufacturer of the CPU is Intel Corporation, located in Santa Clara, CA, USA; the manufacturer of the operating system is Microsoft Corporation, located in Redmond, WA, USA.

5.2. Experiment Setup

The semantics of the experimental locations are classified into 10 types: transportation, education, work, home, industry, healthcare, food, entertainment, shopping, and outdoor. More details are shown in Table 1. Table 4 displays the specific experiment parameters. The anonymity level k ranges from 3 to 15.

Table 4. Experimental parameters.

Parameter	Detail
$\delta_{spa}/\delta_{time}/\delta_{sem}$	0.5/0.7/0.5
δ_{area}/k	0.5/3–15
Location semantics	Transportation, Education, Work, Home, Industry, Healthcare, Food, Entertainment, Shopping, Outdoor

We chose three classical methods to compare with PP-TPS:

- (1) Random scheme [49]: The Random method randomly selects $k - 1$ locations from the candidate locations to generate dummy trajectories.

- (2) MTPPA scheme [50]: The MTPPA method filters the historic trajectories based on the user's stopping point and movement rate. Subsequently, it selects $k - 1$ historic trajectories to compose a k -anonymous set.
- (3) LRM scheme [7]: The LRM method measures the probabilistic and geographic similarity between pairs of locations and utilizes this information to select sampled historic locations. These locations are then categorized into different equivalent probability groups. Afterwards, the scheme merges sampled locations from each equivalent probability class to create a trajectory.

5.3. Experimental Results

5.3.1. Anonymous Success Rate Analysis

In the context of privacy protection within location-based services, a higher level of anonymity ensures that a user's query location is more effectively concealed. This makes it harder for attackers to figure out the user's coordinates, resulting in better privacy protection. This translates to a higher degree of privacy protection. Conversely, a lower degree of anonymity implies a lower level of privacy protection. In this paper, we analyze the privacy protection degree using the anonymity success rate, depicted in Figure 5. The figure illustrates the anonymity success rates for various numbers of mobile users across varying anonymity privacy degrees.

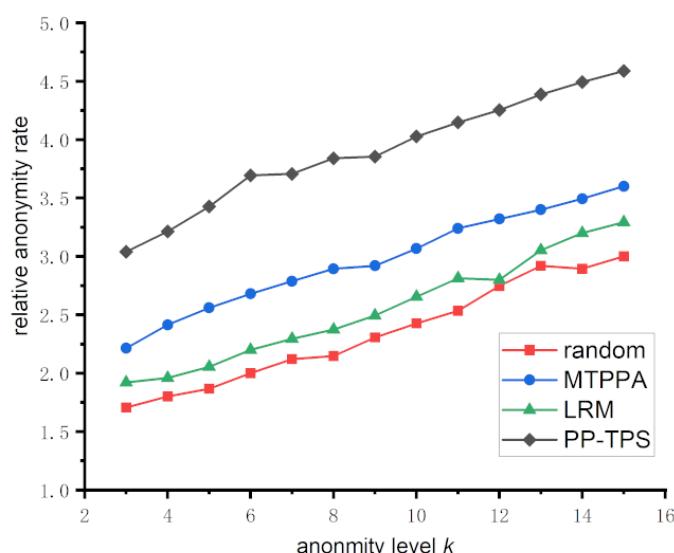


Figure 5. Average anonymity success rate of different anonymity degrees.

The anonymity success rate is a crucial metric for evaluating the effectiveness of location anonymity algorithms. A higher anonymity success rate corresponds to a more effective anonymity algorithm. Formally expressed as $SR = \left| \frac{S'}{S} \right|$, the anonymization success rate represents the percentage of messages that are successfully anonymized out of all anonymization requests made by mobile users.

The vertical coordinate (anonymization success rate) and the horizontal coordinate (anonymity degree k) in Figure 5 exhibit the anonymization success rates of several methods under various anonymity degrees. The Random method exhibits a notably low anonymization success rate due to the easily recognizable randomly generated trajectories, resulting in the lowest success rate. MTPPA has a mediocre success rate in achieving anonymity. Attackers can recognize actual trajectories through temporal reachability, since this approach does not limit the time range of the anonymous trajectories. Although LRM takes into account more dimensions than MTPPA, the possibility of semantic leaks remains. Our PP-TPS, in contrast, guarantees that each dummy trajectory cannot be identified based on multidimensional similarity, leading to a better anonymization success rate. It also places

restrictions on the area enclosed by the trajectories. Overall, the average anonymization success rate increases with higher levels of anonymity (k), as increasing the anonymized trajectories makes it increasingly challenging for attackers to identify real trajectories.

5.3.2. Relative Anonymity Analysis

Relative anonymity is a measure of the proportion between the covered locations within the anonymous area and the anonymity level k , and this proportion must never be less than 1. In other words, the greater the relative anonymity, the greater the number of users involved, indicating a higher level of privacy protection. Figure 6 displays the trend of relative anonymity with respect to the change in k value. The horizontal axis represents the anonymity level k , while the vertical axis represents the relative anonymity.

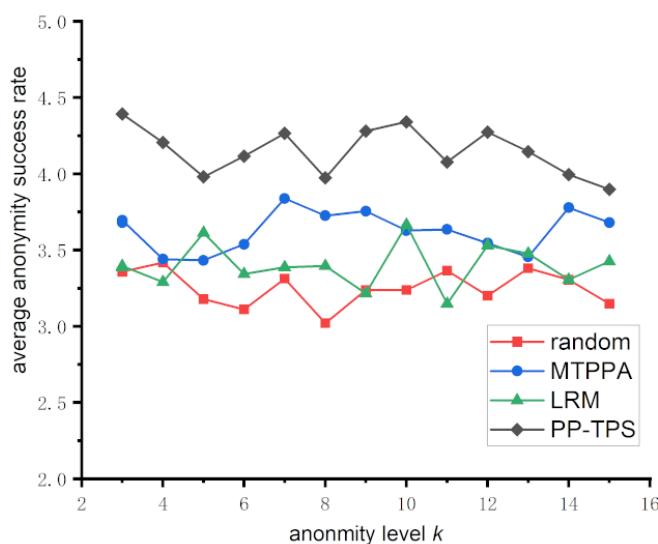


Figure 6. Relative anonymity under different anonymity degrees.

The objective of the Random approach is to randomly choose locations by generating dummy trajectories. Since each location has an equal probability of selection, they all have the same level of recognizability. According to the definition of relative anonymity, when k is fixed, the larger the total number of locations, the greater the relative anonymity. Conversely, when the total number of locations is fixed, increasing the anonymity parameter k decreases relative anonymity. Although the randomized algorithm selects locations randomly, resulting in an unstable size of the anonymity region, the overall trend (average value) in relative anonymity is still relatively low. MTPPA and LRM utilize historic trajectory information to achieve more accurate anonymization. They rely on trajectory similarity or positional similarity, resulting in the formation of larger anonymized regions and a higher relative anonymity compared to the Random scheme. Our PP-TPS method, in comparison to the others, not only focuses on the similarity of location points but also filters historic sub-trajectories by the area enclosed by the sub-trajectories, ensuring a better accomplishment of k -anonymity and aligning with the presented experimental results.

5.3.3. Data Availability Analysis

In this paper, we use the available data rate (AD) to evaluate the number of available data in the anonymized set, which is computed as follows: $AD = \frac{\text{num_reach}(k_{\text{set}})}{\text{num_all}(k_{\text{set}})}$, denoting the number of reachable trajectories in the k -anonymity set over the number of available trajectories in the upper k -anonymity set.

Figure 7 assesses the available data rate across the four approaches. It can be observed that the data availability rate is 1 for all approaches except the Random method, which varies depending on the trajectory anonymity level k . This discrepancy is because the other schemes rely on historic trajectories and generate real trajectories using different algorithms.

It is important to note that the data availability rate for the Random method falls below 1. This is attributed to the random selection of locations for real trajectory generation in this scheme, which may include points in inaccessible areas, such as the center of a large river, where location points cannot be generated. Our PP-TPS approach filters location points based on historic track protection, which are real user-generated location points containing real information, all of which are available data.

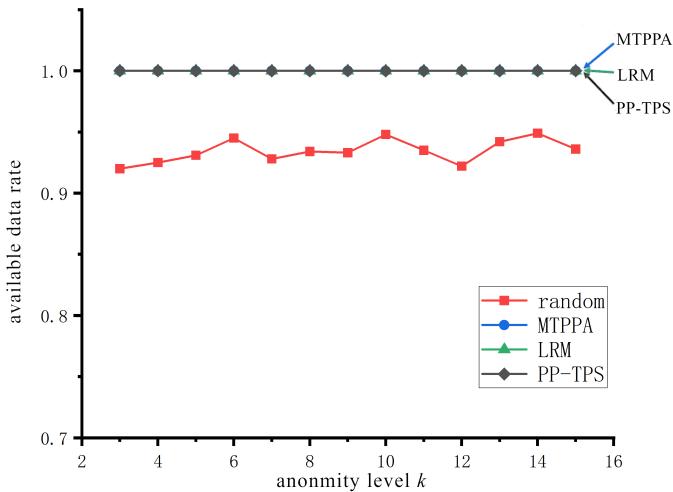


Figure 7. Available data rates at different anonymity levels.

5.3.4. Location Entropy Analysis

The effectiveness of positional privacy protection in trajectory semantics can be evaluated by using location entropy [51], which measures the effectiveness of positional privacy protection. When positional entropy is high, the trajectory semantics become more alike. The formula used to calculate positional entropy is as follows: $H_p = -\frac{1}{N} \sum_{i=1}^N \log_2 \left(\frac{n_i}{T} \right)$. Among these variables, N represents the number of trajectory positions, n_i represents the frequency of semantics at position i , and T represents the total frequency of all semantics in the text. Figure 8 displays the experimental comparison of various anonymity levels.

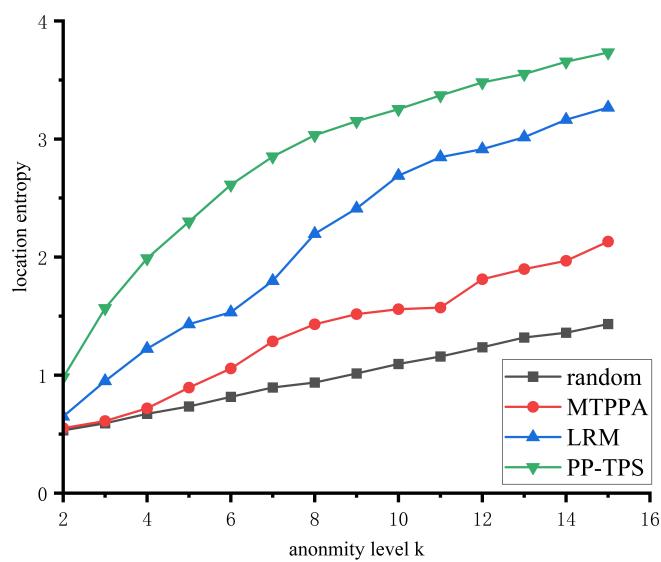


Figure 8. Location entropy at different anonymity levels.

As the anonymity level increases, the position entropy of the three algorithms increases, and position privacy protection intensifies. Our algorithm takes into account the semantic

similarity of trajectories, making it impossible for attackers to identify the true trajectory through semantics. This results in a significantly higher position entropy compared to the other two algorithms, making our algorithm superior. Among these methods, MTPPA employs historic trajectories as virtual trajectories, increasing the likelihood of matching the semantics of actual trajectories compared to random algorithms. Consequently, MTPPA also exhibits a higher position entropy than random algorithms.

5.3.5. Anonymous Location Runtime Analysis

Moreover, we compare the running times of Random, MTPPA, LRM, and PP-TPS, also called the anonymous trajectory generation time. This is because the other operational steps involved in the three schemes are largely similar and require similar times. The experiments compare the anonymous trajectory generation times of Random, MTPPA, LRM, and PP-TPS. The horizontal coordinate represents the number of anonymous trajectories, denoted as the degree of anonymity k . It is clear that when k rises, the time required by the three strategies grows in distinct ways. Random takes the shortest time, since it directly and randomly produces anonymous trajectories. PP-TPS requires additional processing for calculating the similarity of location points and the area enclosed by the sub-trajectory. However, it primarily operates on location points and traverses the length of the real trajectory only twice to calculate the sub-trajectory area. Therefore, it has a comparatively shorter running time compared to LRM and MTPPA. These findings indicate that PP-TPS can reduce computational effort while enhancing trajectory privacy. Figure 9 represents the generation time for anonymized locations across different values of k .

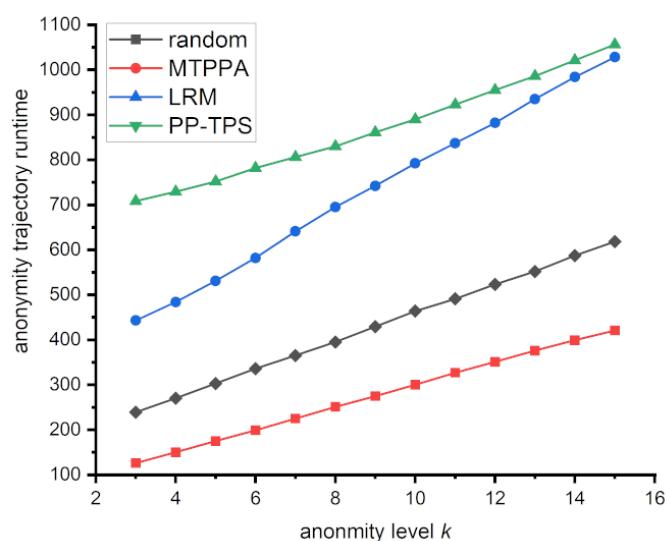


Figure 9. Anonymous Location Runtime at different anonymity levels.

6. Conclusions

This paper proposes an efficient method, PP-TPS, to realize the k -anonymity of a trajectory. PP-TPS involves filtering historic location points by calculating spatial, temporal, and semantic similarities. Based on the discovered similar points, PP-TPS can extract a large amount of corresponding sub-trajectories from many historic trajectories. Then, PP-TPS sifts through these historic sub-trajectories further according to the area distance between them and uses the sub-trajectories of the real trajectory to obtain candidate sub-trajectory sets. Finally, PP-TPS generates $k - 1$ dummy trajectories for the real trajectory based on the candidate sub-trajectory sets.

PP-TPS guarantees that the generated dummy trajectory is similar to the real trajectory at the point level by carefully considering the multidimensional similarity of the location points. Furthermore, it measures the similarity between sub-trajectories using the area distance. This novel approach provides a more straightforward and efficient way to

handle data when compared to computing the similarity of the entire trajectory, making the solution more practical and effective. Overall, PP-TPS surpasses traditional privacy methods, reduces computing costs, and achieves a satisfactory balance. Additionally, the method introduces a novel metric for trajectory similarity based on area. Moreover, PP-TPS utilizes real historic sub-trajectories to generate dummy trajectories, making them more authentic and providing better privacy protection for real trajectories. Although our approach has achieved some success in protecting user privacy, there are still some areas for improvement. Specifically, the challenges we faced included the sparsity of user historic trajectories, which made it relatively difficult to generate a sufficient number of virtual trajectories in some scenarios; at the same time, we also need to address the challenge of meeting the individual needs of different users, which requires us to further optimize the model to be more flexible in meeting user expectations in different scenarios.

In future work, we plan to extend the functionality of the system to provide enhanced privacy protection. This includes work in the following areas: (1) We will think about the possibility of filtering trajectories by sub-trajectory similarity in more dimensions and explore different distance computation methods in multiple dimensions to investigate more practical privacy-preserving solutions for trajectories. (2) Our emphasis will be on exploring the privacy-preserving impact of the algorithm when dealing with sparsely populated historic trajectories. Our objective is to tackle the challenges presented by sparse data to guarantee robust privacy preservation. (3) Our upcoming endeavors involve the creation of a personalized trajectory privacy protection model. This model will empower users to autonomously define the anonymization region for location points and determine the level of trajectory anonymization based on their unique privacy requirements. This approach aims to cater to the varied needs of distinct users and diverse scenarios.

Author Contributions: Conceptualization, H.S. and Y.W.; methodology, H.S. and Y.W.; validation, H.S., Y.W. and M.Z.; writing—original draft preparation, Y.W.; writing—review and editing, H.S.; supervision, M.Z.; funding acquisition, M.Z. and H.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China (61702168, 62072134, U2001205), in part by the Natural Science Foundation of Guangxi Province (2019JJD170020), and in part by the Green Industry Technology Leading Program of Hubei University of Technology (XJ2021000901).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors would like to acknowledge the anonymous reviewers for their useful comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ming, Y.; Yu, X. Efficient Privacy-Preserving Data Sharing for Fog-Assisted Vehicular Sensor Networks. *Sensors* **2020**, *20*, 514. [[CrossRef](#)] [[PubMed](#)]
2. Hasan, A.S.M.T.; Qu, Q.; Li, C.; Chen, L.; Jiang, Q. An Effective Privacy Architecture to Preserve User Trajectories in Reward-Based LBS Applications. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 53. [[CrossRef](#)]
3. Asad, M.; Aslam, M.; Jilani, S.F.; Shaukat, S.; Tsukada, M. SHFL: K-Anonymity-Based Secure Hierarchical Federated Learning Framework for Smart Healthcare Systems. *Future Internet* **2022**, *14*, 338. [[CrossRef](#)]
4. Wu, W.; Shang, W.; Lei, R.; Yang, X. A Trajectory Privacy Protect Method Based on Location Pair Reorganization. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 8635275. [[CrossRef](#)]
5. Zhang, X.; Lu, R.; Ray, S.; Shao, J.; Ghorbani, A.A. Spatio-temporal similarity based privacy-preserving worker selection in mobile crowdsensing. In Proceedings of the 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 7–11 December 2021; pp. 1–6.
6. Sun, G.; Cai, S.; Yu, H.; Maharjan, S.; Chang, V.; Du, X.; Guizani, M. Location privacy preservation for mobile users in location-based services. *IEEE Access* **2019**, *7*, 87425–87438. [[CrossRef](#)]

7. Wang, Y.; Li, M.; Luo, S.; Xin, Y.; Zhu, H.; Chen, Y.; Yang, G.; Yang, Y. LRM: A location recombination mechanism for achieving trajectory k -anonymity privacy protection. *IEEE Access* **2019**, *7*, 182886–182905. [[CrossRef](#)]
8. Zhang, J.; Li, Y.; Ding, Q.; Lin, L.; Ye, X. Successive Trajectory Privacy Protection with Semantics Prediction Differential Privacy. *Entropy* **2022**, *24*, 1172. [[CrossRef](#)]
9. Du, Y.; Hu, Y.; Zhang, Z.; Fang, Z.; Chen, L.; Zheng, B.; Gao, Y. LDPTTrace: Locally Differentially Private Trajectory Synthesis. *arXiv* **2023**, arXiv:2302.06180.
10. Mohammadi, S.; Darestani, S.A.; Vahdani, B.; Alinezhad, A. A robust neutrosophic fuzzy-based approach to integrate reliable facility location and routing decisions for disaster relief under fairness and aftershocks concerns. *Comput. Ind. Eng.* **2020**, *148*, 106734. [[CrossRef](#)]
11. Qiu, G.; Shen, Y.; Cheng, K.; Liu, L.; Zeng, S. Mobility-aware privacy-preserving mobile crowdsourcing. *Sensors* **2021**, *21*, 2474. [[CrossRef](#)]
12. Kim, J.W.; Jang, B. Deep learning-based privacy-preserving framework for synthetic trajectory generation. *J. Netw. Comput. Appl.* **2022**, *206*, 103459. [[CrossRef](#)]
13. Komishani, E.G.; Abadi, M.; Deldar, F. PPTD: Preserving personalized privacy in trajectory data publishing by sensitive attribute generalization and trajectory local suppression. *Knowl.-Based Syst.* **2016**, *94*, 43–59. [[CrossRef](#)]
14. Tu, Z.; Zhao, K.; Xu, F.; Li, Y.; Su, L.; Jin, D. Beyond k -anonymity: Protect your trajectory from semantic attack. In Proceedings of the 2017 14th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), San Diego, CA, USA, 12–14 June 2017; pp. 1–9.
15. Zhao, Y.; Luo, Y.; Yu, Q.; Hu, Z. A privacy-preserving trajectory publication method based on secure start-points and end-points. *Mob. Inf. Syst.* **2020**, *2020*, 3429256. [[CrossRef](#)]
16. Zhou, K.; Wang, J. Trajectory protection scheme based on fog computing and K -anonymity in IoT. In Proceedings of the 2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS), Matsue, Japan, 18–20 September 2019; Volume 10, pp. 1–6.
17. Li, B.; Zhu, H.; Xie, M. Releasing Differentially Private Trajectories with Optimized Data Utility. *Appl. Sci.* **2022**, *12*, 2406. [[CrossRef](#)]
18. Guerra-Balboa, P.; Pascual, A.M.; Parra-Arnau, J.; Forné, J.; Strufe, T. Anonymizing Trajectory Data: Limitations and Opportunities. In Proceedings of the Third AAAI Workshop on Privacy-Preserving Artificial Intelligence (PPAI-22), Virtual, 28 February 2022; AAAI Press (Association for the Advancement of Artificial Intelligence): Washington, DC, USA, 2022.
19. Yu, W.; Shi, H.; Xu, H. A Trajectory K -Anonymity Model Based on Point Density and Partition. *arXiv* **2023**, arXiv:2307.16849.
20. Zhang, A.; Li, X. Research on privacy protection of dummy location interference for Location-Based Service location. *Int. J. Distrib. Sens. Netw.* **2022**, *18*, 15501329221125111. [[CrossRef](#)]
21. Yang, C.; Yan, W. False Trajectory Privacy Protection Scheme Based on Location Service. In *IOP Conference Series: Materials Science and Engineering*; IOP Publishing: Bristol, UK, 2020; Volume 768, p. 072025.
22. Guo, K.; Wang, D.; Zhi, H.; Gao, Y.; Lu, Y. Privacy-preserving Trajectory Generation Algorithm Considering Utility based on Semantic Similarity Awareness. In Proceedings of the ICC 2022-IEEE International Conference on Communications, Gangnam-gu, Republic of Korea, 16–20 May 2022; pp. 992–997.
23. Furtado, A.S.; Kopanaki, D.; Alvares, L.O.; Bogorny, V. Multidimensional similarity measuring for semantic trajectories. *Trans. GIS* **2016**, *20*, 280–298. [[CrossRef](#)]
24. Sweeney, L. k -anonymity: A model for protecting privacy. *Int. J. Uncertain. Fuzziness -Knowl.-Based Syst.* **2002**, *10*, 557–570.
25. Gruteser, M.; Grunwald, D. Anonymous usage of location-based services through spatial and temporal cloaking. In Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, San Francisco, CA, USA, 5–8 May 2003; pp. 31–42.
26. Chen, C.; Hu, X.; Li, Y.; Tang, Q. Optimization of Privacy Budget Allocation In Differential Privacy-Based Public Transit Trajectory Data Publishing for Smart Mobility Applications. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 15158–15168. [[CrossRef](#)]
27. Jiang, J.; Han, G.; Wang, H.; Guizani, M. A survey on location privacy protection in wireless sensor networks. *J. Netw. Comput. Appl.* **2019**, *125*, 93–114. [[CrossRef](#)]
28. Pingley, A.; Zhang, N.; Fu, X.; Choi, H.; Subramaniam, S.; Zhao, W. Protection of query privacy for continuous location based services. In Proceedings of the 2011 IEEE INFOCOM, Shanghai, China, 10–15 April 2011; pp. 1710–1718.
29. Wu, Q.; Liu, H.; Zhang, C.; Fan, Q.; Li, Z.; Wang, K. Trajectory protection schemes based on a gravity mobility model in IoT. *Electronics* **2019**, *8*, 148. [[CrossRef](#)]
30. Lu, R.; Lin, X.; Shi, Z.; Shao, J. PLAM: A privacy-preserving framework for local-area mobile social networks. In Proceedings of the IEEE INFOCOM 2014-IEEE Conference on Computer Communications, Toronto, ON, Canada, 27 April–2 May 2014; pp. 763–771.
31. Niu, B.; Zhu, X.; Li, Q.; Chen, J.; Li, H. A novel attack to spatial cloaking schemes in location-based services. *Future Gener. Comput. Syst.* **2015**, *49*, 125–132. [[CrossRef](#)]
32. Kuang, L.; Wang, Y.; Zheng, X.; Huang, L.; Sheng, Y. Using location semantics to realize personalized road network location privacy protection. *EURASIP J. Wirel. Commun. Netw.* **2020**, *1*, 1–16. [[CrossRef](#)]
33. Gao, S.; Jianfeng, M.; Weisong, S.; Guoxing, Z.; Cong, S. TrPF: A trajectory privacy-preserving framework for participatory sensing. *IEEE Trans. Inf. Forensics Secur.* **2013**, *8*, 874–887. [[CrossRef](#)]

34. Zhao, D.; Song, G.; Jin, Y.; Wang, X. Query probability-based location privacy protection approach. *J. Comput. Appl.* **2017**, *37*, 347.
35. Wang, B.; Liu, J.; Dai, L. K-Anonymity-Based Privacy-Preserving and Efficient Location-Based Services for Internet of Vehicles Withstand Viterbi Attack. In Proceedings of the International Conference on Image, Vision and Intelligent Systems 2022 (ICIVIS 2022), Jinan, China, 15–17 August 2022; pp. 1016–1028.
36. Du, Y.; Cai, G.; Zhang, X.; Liu, T.; Jiang, J. An Efficient Dummy-Based Location Privacy-Preserving Scheme for Internet of Things Services. *Information* **2019**, *10*, 278. [[CrossRef](#)]
37. Niu, B.; Qinghua, L.; Xiaoyan, Z.; Guohong, C.; Hui, L. Achieving k-anonymity in privacy-aware location-based services. In Proceedings of the IEEE INFOCOM 2014–IEEE Conference on Computer Communications, Toronto, ON, Canada, 27 April–2 May 2014; pp. 754–762.
38. Yang, D.; Ye, B.; Chen, Y.; Zhou, H.; Qian, X. A Dummy Location Selection Algorithm Based on Location Semantics and Physical Distance. In Proceedings of the Information Security Practice and Experience: 16th International Conference, ISPEC 2021, Nanjing, China, 17–19 December 2021; Proceedings *16*, pp. 283–295.
39. Chen, S.; Shen, H. Semantic-aware dummy selection for location privacy preservation. In Proceedings of the 2016 IEEE Trustcom/BigDataSE/ISPA, Tianjin, China, 23–26 August 2016; pp. 752–759.
40. Tan, R.; Tao, Y.; Si, W.; Zhang, Y.-Y. Privacy preserving semantic trajectory data publishing for mobile location-based services. *Wirel. Netw.* **2020**, *26*, 5551–5560. [[CrossRef](#)]
41. Xing, L.; Zhang, D.; Wu, H.; Ma, H.; Zhang, X. Distributed K-Anonymous Location Privacy Protection Algorithm Based on Interest Points and User Social Behavior. *Electronics* **2023**, *12*, 2446. [[CrossRef](#)]
42. Wang, S.; Chen, C.; Zhang, G. Similarity-based privacy protection for publishing k-anonymous trajectories. *Front. Comput. Sci.* **2022**, *16*, 163605. [[CrossRef](#)]
43. Qian, J.; Jiang, H.; Yu, Y.; Wang, H.; Miao, D. Multi-level personalized k-anonymity privacy-preserving model based on sequential three-way decisions. *Expert Syst. Appl.* **2023**, *239*, 122343. [[CrossRef](#)]
44. Gao, S.; Ma, J.; Sun, C.; Li, X. Balancing trajectory privacy and data utility using a personalized anonymization model. *J. Netw. Comput. Appl.* **2014**, *38*, 125–134. [[CrossRef](#)]
45. Zhang, M.; Li, X.; Miao, Y.; Luo, B.; Ren, Y.; Ma, S. PEAK: Privacy-Enhanced Incentive Mechanism for Distributed K-Anonymity in LBS. *IEEE Trans. Knowl. Data Eng.* **2023**. [[CrossRef](#)]
46. Li, F.; Liao, L.; Zhang, L.; Zhu, X.; Zhang, B.; Wang, Z. An Efficient Approach for Measuring Semantic Similarity Combining WordNet and Wikipedia. *IEEE Access* **2020**, *8*, 184318–184338. [[CrossRef](#)]
47. Yang, Z.; Yang, T.; Chen, K.; Wei, F. Real-time calculation of enclosed area of mouse trajectory in B-ultrasound diagnostic instrument. *J. Guangdong Polytech. Norm. Univ.* **2007**, *12*, 5–7.
48. Zheng, Y.; Li, Q.; Chen, Y.; Xie, X.; Ma, W.-Y. Understanding mobility based on GPS data. In Proceedings of the 10th International Conference on Ubiquitous Computing, Seoul, Republic of Korea, 21–24 September 2008; pp. 312–321.
49. Liu, H.; Li, X.; Li, H.; Ma, J.; Ma, X. Spatiotemporal correlation-aware dummy-based privacy protection scheme for location-based services. In Proceedings of the IEEE INFOCOM 2017–IEEE Conference on Computer Communications, Atlanta, GA, USA, 1–4 May 2017; pp. 1–9.
50. Xu, Z.; Zhang, J.; Tsai, P.-W.; Lin, L.; Zhuo, C. Spatiotemporal mobility based trajectory privacy-preserving algorithm in location-based services. *Sensors* **2021**, *21*, 2021. [[CrossRef](#)]
51. Sun, Y.; Chen, M.; Hu, L.; Qian, Y.; Hassan, M.M. ASA: Against statistical attacks for privacy-aware users in location based service. *Future Gener. Comput. Syst.* **2017**, *70*, 48–58. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.