Journal of Spatial Information Science
Number 13 (2016), pp. 61–99
doi:10.5311/JOSIS.2016.13.263

Review Article

Trajectory data mining: A review of methods and applications

Jean Damascène Mazimpaka and Sabine Timpf

Department of Geography, University of Augsburg, Germany

Received: October 10, 2015; returned: January 27, 2016; revised: July 7, 2016; accepted: September 6, 2016.

Abstract: The increasing use of location-aware devices has led to an increasing availability of trajectory data. As a result, researchers devoted their efforts to developing analysis methods including different data mining methods for trajectories. However, the research in this direction has so far produced mostly isolated studies and we still lack an integrated view of problems in applications of trajectory mining that were solved, the methods used to solve them, and applications using the obtained solutions. In this paper, we first discuss generic methods of trajectory mining and the relationships between them. Then, we discuss and classify application problems that were solved using trajectory data and relate them to the generic mining methods that were used and real world applications based on them. We classify trajectory-mining application problems under major problem groups based on how they are related. This classification of problems can guide researchers in identifying new application problems. The relationships between the methods together with the association between the application problems and mining methods can help researchers in identifying gaps between methods and inspire them to develop new methods. This paper can also guide analysts in choosing a suitable method for a specific problem. The main contribution of this paper is to provide an integrated view relating applications of mining trajectory data and the methods used.

Keywords: trajectories, mining methods, mining application problems, applications

1 Introduction

The development in the field of information and communication technology especially in mobile sensing and wireless communication is flooding us with data containing time-varying geographic positions. Though this kind of data is also associated with challenges such as exhausting our storage capacity and our data transmission bandwidth, researchers

have shown that these data sets constitute a precious resource. Their analysis can lead to solutions for important research problems in different fields such as urban planning, transportation, behavioral ecology, sports scene analysis, surveillance, and security [53].

Researchers in movement analysis have made important contributions by developing methods and tools to solve specific application problems. As researchers continue to address diverse application problems [73] we urgently need an integrated view of the application problems solved using trajectory data and the mining methods available for addressing each class of application problem. Such an integrated view would allow other researchers to easily identify the issues already solved, the methodological gaps, and to draw inspiration for new method development from the relations between problems and methods. The objective of this paper is to contribute to the provision of an integrated view of trajectory mining application problems and methods. The paper also presents a survey of applications that exploit the solutions to these application problems. Available analysis methods can be classified as computational, statistical, visual, or a combination of these. This paper focuses on computational methods of trajectory mining.

The remainder of this paper is organized as follows: after this introduction, Section 2 presents a brief overview of trajectory data and trajectory data mining. Section 3 discusses mining methods for trajectory data while Section 4 discusses application problems addressed using these methods. In addition, Section 4 relates the mining application problems to the methods presented in the paper, which are used to solve them. Section 5 reviews applications that exploit the solutions to application problems discussed in Section 4. Section 6 briefly presents open issues that need to be addressed for future development of trajectory data mining. In Section 7, we conclude and list directions for future work.

2 Trajectory data and trajectory data mining

In this section, we briefly explain the background concepts that form the basis of the research surveyed. We define the concept of trajectory data and locate trajectory data mining in the broader domain of data mining. For reasons of space, we do not discuss these concepts in detail but interested readers can follow the corresponding references indicated in this section.

2.1 Trajectory data

Based on the technology by which they are recorded, mobility data is available in different forms. Spinsanti et al. [122] differentiated GPS (global positioning system), GSM (global system for mobile communications), and geo-social network based trajectory data. Pelekis and Theodoris [112] added two other forms; RFID (radio frequency identification) based and Wi-Fi based data. GPS based data is composed of temporally ordered sequences of geographic coordinates recorded by a GPS-enabled device carried by the moving object. GSM based data is composed of temporally ordered sequences of identifiers of the cells in which the moving object passes. Geo-social network based data is content found on Internet social media and to which geographic coordinates have been attached. RFID based data contain a sequence of identifiers of RFID readers through which the moving object passed, while Wi-Fi based data contain a sequence of identifiers of access points that communicated with the moving object. Though the properties of these forms of data, e.g., their accuracy, show that they are very different [112, 122], they have been used to address similar or related application problems using similar or related mining methods.

A trajectory can be generally formally represented as: T=$\langle p_1 ... p_n \rangle$ where p_k=(id_k, loc_k, t_k, A_k) is the k^{th} position, id_k is the position identifier, loc_k is the spatial location of the position, t_k is the time at which the position was recorded, and A_k is a possibly empty list of additional descriptive data (e.g., direction, occupancy status, etc.). The spatial location of the position may be represented in different ways depending on the recording technology (e.g., loc_k= (x, y) for GPS-based data, and loc_k = *cell ID* for GSM-based data).

2.2 Trajectory data mining

Data mining is an important step of a process, commonly known as *knowledge discovery* [37, 95] that extracts useful information from huge datasets. Data mining methods and applications have been widely surveyed in the general data mining domain. For instance, a survey of data mining methods for classical relational and transactional data can be found in [37] and [57]. Beyond the classical relational and transactional data, some researchers [100–102] provided an overview of mining tasks on geographic data. The tasks they discuss are generally analogous to the tasks in the general data mining domain but the mining methods are adapted to handle the peculiarity of geographic data such as the spatial dependency and the combination of spatial and non-spatial components. Though the tasks presented in these surveys have analogous tasks in the movement data domain, the review does not consider the temporal dimension that is inherent in movement data.

Andrienko et al. [5] proposed a conceptual framework for classifying movement analysis techniques in general. In their framework, they consider a movement analysis process to consist of one or more tasks. The types of tasks are distinguished based on the type of information sought, which is defined in terms of the elements involved, i.e., objects, space or time, and the specific target which is either characteristic or relation. They also distinguish tasks based on the analysis level, which is either elementary or synoptic. Andrienko et al. [5] also proposed a taxonomy of methods for analyzing movement data. This taxonomy focuses on the methods in the domain of visualization and interaction. The contribution of our paper extends the work of Andrienko et al. [5] into the domain of computational analysis methods.

From the applications side, Castro et al. [22] surveyed the work on mining traces of taxis. They classified the surveyed work into three categories: social dynamics, traffic dynamics, and operational dynamics. Though their survey provided an extensive literature review, it considered only the work done on taxi traces. We therefore need a survey that considers applications of trajectory data mining without limit to a specific type of trajectories. Our work differs from the survey in [22] by considering traces in general and by discussing specific application problems that are of major interest in different application fields.

We survey the work done from the application side according to the framework shown in Figure 1. As the framework shows, there are different types of moving objects that can be tracked. Tracking these objects generates a set of trajectories and some questions of which the solutions can be used in some application fields. We call these questions application problems because their solutions are not application in themselves but knowledge used in application fields. The application problems are solved using specific trajectory mining methods. For instance, if moving people are tracked we get their trajectories and some

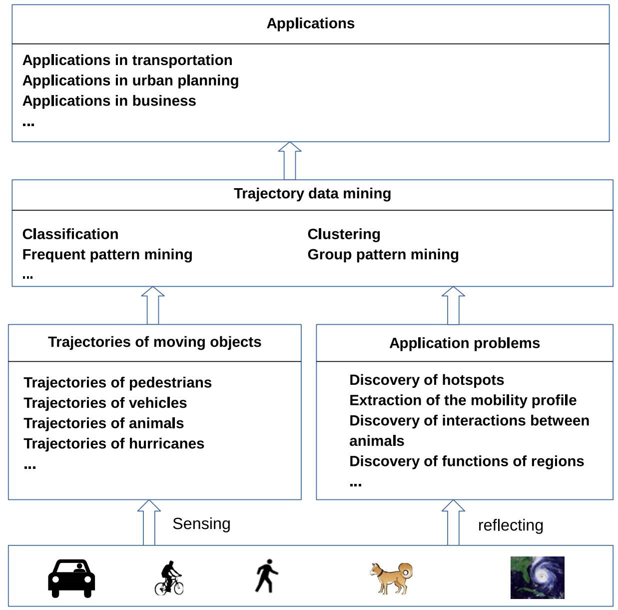

Figure 1: General framework of application-driven trajectory data mining.

questions come to our mind regarding, for instance, the characteristics of these people or the geographical space where they move. By mining these trajectories we may discover places where many people stop frequently and stay for a considerable long duration. These places are characterized as hotspots and the knowledge about hotspots can be used, for instance, in business for advertisement, in public security for crowd monitoring, and in transportation for taxi pick-up point recommendation. Therefore, hotspots discovery is an application problem of which the solution can have various applications.

The core task at the mining step is the application of a specific mining method. This is preceded by a trajectory data pre-processing step, which prepares the data for mining. This step includes tasks such as data cleaning, trajectory compression, map matching, and trajectory segmentation. Data cleaning aims at filtering noise from the data while trajectory compression aims at reducing the amount of trajectory sample points to reduce the volume of data to be processed. Map matching aims at adjusting trajectory sample points to the roads in case the movement follows a road network while trajectory segmentation divides a trajectory into meaningful sub-trajectories required for subsequent operations. A discussion of trajectory pre-processing tasks can be found in [150].

3 Trajectory mining methods

Similar to the general domain of data mining, trajectory data mining aims at discovering interesting knowledge (e.g., patterns) from the data and it has two primary goals: prediction and description. Prediction consists in using some variables in the data to determine unknown or future values of other variables of interest, while description focuses on finding human-interpretable structures describing the data. We do not try to discuss all existing mining methods; instead we review generic ones and identify the relationships between them. By generic mining methods we mean mining methods applied on a wide range of trajectory data in a wide range of applications. We divide the methods discussed into two categories: primary methods and secondary methods. Given a trajectory dataset, primary methods aim at categorizing the trajectories based on their properties. On the other hand, secondary mining methods are composite methods in the sense that they apply a sequence of primary mining methods or classical statistical methods or a combination of the two.

The discussion of the methods focuses on the objective of the method, its general working principle and the data granularity level at which it works (whole trajectories versus subtrajectories, single trajectory versus multiple trajectories, and trajectories of single moving object versus trajectories of multiple objects). The discussion ends with an attempt to identify relationships between different methods.

3.1 Primary mining methods

Primary trajectory mining methods are used to categorize the trajectories based on their properties. The application of a primary mining method followed by the interpretation of its results may be the sole intended analysis or the primary mining method may be used to prepare the data for an extended analysis possibly with other mining methods. Primary trajectory mining methods include trajectory clustering and trajectory classification.

3.1.1 Clustering

Trajectory data clustering aims at describing a trajectory dataset by grouping the trajectories into a finite set of categories, also called clusters, based on their movement characteristics. The trajectories in the same cluster exhibit movement characteristics that are similar and different from those of trajectories in other clusters.

There have been efforts to develop trajectory-specific clustering approaches mainly by adapting statistical and probabilistic models to account for the characteristics of trajectories. For example, the mixture models based clustering approach proposed by Gaffney and Smyth [42] groups together trajectories that are likely to be generated by a common representative trajectory by adding Gaussian noise. In the same line, Alon et al. [2] model trajectories as sequences of transitions between positions and use a hidden Markov model (HMM) that best fits the trajectories to model a cluster.

Though trajectory-specific clustering approaches exist, state-of-the-art clustering algorithms for trajectories are extensions of traditional clustering algorithms through a proper definition of similarity (or distance) functions. A survey of traditional clustering algorithms

can be found in [57]. Depending on the goal of analysis some similarity (or distance) function such as similar route, similar destination, similar source, similar route and destination, or similar direction is used to determine which trajectories belong to the same cluster. A detailed discussion of how distance and similarity functions are applied to determine cluster membership can be found in Rokach [119].

Among the algorithms that have been extended, two well-known can be mentioned: DBSCAN (density based spatial clustering of applications with noise) [36] and OPTICS (ordering points to identify the clustering structure) [6]. As examples of such extensions, T-OPTICS [106] was developed as an extension of OPTICS by defining a spatio-temporal distance for comparing and clustering trajectories while ST-DBSCAN (spatial-temporal DBSCAN) [14] uses two parameters for similarity measure to improve the identification of clusters and noise.

Several clustering algorithms have been developed and categorized in different ways (e.g., see in [40, 57, 58]) based on different features but a crisp boundary between the categories is difficult to draw. Most of the categorization approaches identify partitioning, hierarchical, and density-based algorithms. Partitioning algorithms group all objects into a pre-specified number of clusters. These algorithms start from a random partitioning which is then refined through iterations that may move an object from one cluster to another. K-means [94] is an example of such algorithms. Hierarchical algorithms organize the objects in a multilevel structure of clusters and sub-clusters. Using loose proximity requirements, the algorithms find clusters at a higher level while by tightening the proximity requirements they find sub-clusters. An example of such an algorithm is the BIRCH (balanced iterative reducing and clustering using hierarchies) [147]. Density-based algorithms partition the objects into clusters based on their density, i.e., starting from one object, the cluster grows as long as new objects exist in the neighborhood, and the cluster is considered valid if its total number of objects exceeds a threshold. DBSCAN is an example of algorithm in this category.

Trajectory clustering works on a set of trajectories taking each trajectory in turn and assigning it to a cluster. However, there are some cases in which the input to the clustering operation is not a set of trajectories but only one trajectory. In these cases, the objective is to cluster the points of the single trajectory with the aim of characterizing some positions on it. This case is exemplified in Palma et al. [109] where the points on the input trajectory are clustered to discover stops on it.

Trajectory clustering can be applied on either whole trajectories or sections of trajectories depending on the goal of analysis and the similarity function applied. For example, if the similarity function is defined as similar origin and similar destination, the trajectories can be clustered as a whole because the route followed is not important. An example of this case can be seen in [43] where the interest is on the overall directions of extra-tropical cyclone trajectories for obtaining clusters such as *south-to-north oriented* and *west-to-east oriented*. On the other hand, if the interest is on different locations traversed by the trajectories, for example the similarity is defined as having visited the same types of places; the clustering is applied on sections of trajectories. The TraClus clustering algorithm [79] follows this approach.

3.1.2 Classification

The objective of classification is to find a rule to assign objects into pre-defined classes. The knowledge about class assignment is available in the form of a set of predefined classes and a sample set of objects already labeled with the class they belong to. This sample is called a training set. Similar to classification in general, trajectory classification aims at determining the class label of trajectories from a predefined set of labels based on the features of the trajectories. For example, the trajectory classification may be aimed at labeling each trajectory from a large set with its transportation mode, given a small set of trajectories manually labeled with the transportation mode used.

Most trajectory classification algorithms follow a traditional two-step approach: first extracting a set of discriminative features and then using the extracted features to train an existing standard classification model. The first step involves identifying the properties of the trajectories that are likely best for identifying the class to which each trajectory belongs, and determining the values of the properties which are then represented in a specific form (e.g., a vector). For example, properties such as the average speed of the trajectory, average acceleration, trajectory duration, and length can be used. The discriminative power of the considered feature depends on the types of classes at hand. For example, we may consider that the acceleration has a much higher discriminative power than the trajectory length for the vehicle type classes. The second step selects a standard classification model and applies it to the extracted features.

A detailed discussion of the traditional classification algorithms can be found in [57]. Among these methods, the following have been commonly applied in trajectory classification. The decision tree algorithm has been used by Zheng et al. [152] to classify trajectories into different transportation modes. They first segment the trajectories and extract discriminative features such as average velocity of a segment, heading change rate, and velocity change rate, which are then fed into a decision tree based inference model for classification. Support vector machines (SVMs) were used by Bolbol et al. [15] to solve the same issue of transportation mode classification. Through a statistical evaluation, they first analyze the discriminative power of four features (speed, acceleration, distance, and rate of change in heading) on six transportation modes (bus, car, cycle, train, underground, and walk). After identifying the speed and acceleration as the best discriminative features, they applied the SVM algorithm on these features to study equal-sized instances of several segments of trajectories for classifying them.

In many of its applications, trajectory classification is performed after other analysis operations such as segmentation and clustering which prepare the features on which the classification will be based. For example, the TraClass framework [78] applies trajectory segmentation and clustering to extract region and sub-trajectory features that are then used for a support vector machine (SVM) based classification of the trajectories.

3.2 Secondary mining methods

While primary trajectory mining methods aim at categorizing the trajectories, secondary mining methods generally aim at analyzing the spatial, temporal, or spatio-temporal arrangement of the individual trajectories, within their categories or between categories. Several mining methods can be put in this category, but we discuss in this section a few generic ones: pattern mining, outlier detection, and prediction.

3.2.1 Pattern mining

Trajectory pattern mining aims at discovering and describing the movement patterns hidden in trajectories. It provides information about when and where the pattern occurs, and the entities involved in it. A large number of types of movement patterns have been reported in the literature. An integrated view of them can be found in the survey done by Dodge et al. [35]. Likewise, several methods have been developed for mining these patterns. The methods can be put into three categories: repetitive pattern mining, frequent pattern mining, and group pattern mining.

Repetitive pattern mining applies basically on trajectories of a single moving object while frequent pattern mining applies on trajectories of multiple moving objects but in a relative time; that is, the objects may not move at the same time and the only condition is that they visit approximately the same places in the same sequence. Similar to frequent pattern mining, group pattern mining applies to trajectories of multiple objects but the movement is considered in an absolute time; that is, the objects move together.

Repetitive pattern mining concerns regular movement patterns such as the movement of a commuter, which is repeated every day, or the movement of a migratory bird, which is repeated every season. A repetitive pattern is also called a periodic pattern because the object follows approximately the same route after an approximately constant time period. The discovery of periodic patterns is complicated by the approximate nature of the patterns in terms of space and time; i.e., the object does not visit exactly the same location at corresponding time instants of the period and the period does not exactly have the same value in different cycles.

A common approach in discovering periodic movement patterns is to apply the mining on sequences of locations. In an earlier study, Cao et al. [20] required the period to be specified as input and clustered the locations corresponding to the same offset of the period to discover frequent regions. They then iteratively combined discovered frequent regions to get complete periodic patterns. The work presented by Cao et al. [20] extended the approach presented in a previous study [97], but both studies required the period as a user input to the mining process. The *Periodica* algorithm [85] overcame this undesirable requirement to specify the period in advance. This algorithm first detects automatically the period using an approach combining Fourier transformation and autocorrelation. The algorithm finds reference spots, which are dense regions containing more trajectory points than others, and detects the different periods in each. The algorithm then discovers periodic patterns from movement sequences between reference spots with the same period using a hierarchical clustering for which the distance measure is based on a probabilistic model. Compared to most of previous related work, the *Periodica* algorithm presents two features that address the unknown and varying period issue commonly found on real life movement data: automatic period detection and the consideration of different periods during the periodic pattern discovery.

Frequent pattern mining is about extracting (parts of) routes that have been frequently followed by the moving objects in the trajectory dataset. Frequent trajectory patterns can be defined using spatial or spatiotemporal characteristics of the trajectories [68]. The definition based on spatial characteristics considers only the sequence of the locations visited. The *frequent spatiotemporal sequential patterns* [19] and the *generalized sequential patterns* (GSP) [108] are examples of this case. The definition based on spatiotemporal characteristics considers, in addition to the sequence of the locations visited, the transition time between the locations. The *T-Patterns* (or simply *trajectory patterns*) [49] are an example of this

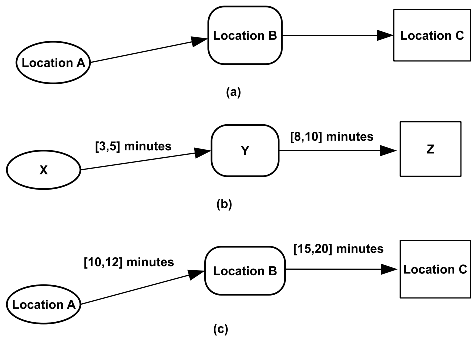

Figure 2: (a) A spatiotemporal sequential pattern, (b) a temporally annotated sequence (TAS), and (c) a T-pattern.

second case. Giannotti et al. [49] defined a T-pattern as "a set of individual trajectories that share the property of visiting the same sequence of places with similar travel times.". T-patterns evolved from the *temporally annotated sequences* or *TAS* introduced by Giannotti et al. [47]. In TAS, the elements of a sequence are generic events, e.g., purchases or logged web accesses, without any specific spatial information. Figure 2 shows three types of sequences that make different frequent patterns: for spatiotemporal sequential patterns (a) only the sequence of locations is considered while for T-patterns (c) the transition time between locations (e.g., from location A to location B it takes between 10 and 12 minutes) is also considered. T-patterns (c) combine the feature of location sequence from spatiotemporal sequential patterns (a) with the feature of transition time from TAS (b).

A common approach to mining frequent patterns consists in finding important regions from the trajectories and then applying sequence mining on a temporally annotated sequence of these regions [49,65]. In this line, Giannotti et al. [49] developed two approaches for extracting T-patterns. The first approach involves a discretization of the space to identify the regions of interest. Then, the actual mining step applies the method introduced in Giannotti et al. [47] to mine sequences of regions of interest with temporal annotations. The second approach dynamically derives the regions of interest from trajectory segments and then translates the trajectories into a set of sequences of regions from which it iteratively builds T-patterns.

Group pattern mining aims at extracting movement patterns involving groups of objects that move together. A general condition is that the objects that form a group stay close in space for a considerable period of time. Several group patterns and their variants have been defined based on the general condition of spatiotemporal closeness, the internal structure of the group and the characteristics of group members. The mostly studied patterns are flocks [12,130], convoy [63], and swarm [84]. Figure 3 shows examples of these three patterns.

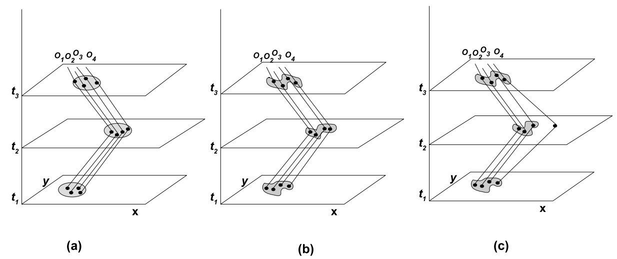

Figure 3: Examples of group pattern: (a) Flock, (b) Convoy, and (c) swarm.

A flock is a group of at least *m* objects that travel together for at least *k* consecutive timestamps such that at any of these timestamps they are found within a disc of radius *r*. The pattern is defined by the three parameters *m*, *r*, and *k*.

The convoy pattern relaxes the disc shape constraint to represent a group of moving objects forming any shape. It is a group of at least *m* objects traveling together for at least *k* consecutive timestamps such that at each of these timestamps the group can be found using a density-based clustering with parameters *d* as the neighborhood distance and *m* as the minimum number of objects.

The swarm pattern is an extension of the convoy pattern, which further relaxes the constraint of consecutive timestamps. It is defined by the same parameters as the convoy except that in a swarm *k* is the minimum number of timestamps at which the group is found irrespective of whether the timestamps are consecutive or not. For example, with m=4 and k=2 the pattern in Figure 3 (c) is not a convoy because only three objects (O_1, O_2, and O_3) are found in the group for at least two consecutive timestamps. However, this pattern is a swarm because at two timestamps (t_1 and t_3) the group includes the required minimum of four objects and the swarm pattern does not require the timestamps to be consecutive.

The category of group patterns also includes patterns in which the members of the group have some interaction in which each member has a specific role. An example of such pattern is the *leadership* pattern [4] in which the group comprises a leader which moves ahead and followers for the time duration of the pattern. Another example is the *chasing* pattern [31] in which the object moving ahead aims at escaping the follower, which tries to reach it. Different examples of group patterns can be found in [76] under the general concept of relative motion (REMO).

Common approaches to group pattern mining involve clustering methods and checking the condition on parameters that define the pattern such as the minimum number of group members and the minimum duration of the pattern. For example, density-based clustering has been applied in [129] to mine flocks, in [62,63] to mine convoys, and in [86,139] to mine swarms. For swarms, the analysis starts by applying a clustering method at the level of a single timestamp, and then links spatial clusters belonging to different timestamps but sharing an appropriate fraction of objects regardless of their temporal distance.

3.2.2 Outlier detection

The objective in trajectory outlier detection is to find trajectories that do not comply with the general behavior of the trajectory dataset. While pattern mining focuses on patterns that are common in the trajectory dataset, outlier detection focuses on rare patterns (e.g., following a path different from the common path followed by most of the other trajectories).

Similar to trajectory classification, trajectory outlier detection can be performed by mining either whole trajectories or parts of trajectories. The work presented in [146] is an example of outlier detection applied on whole trajectories, while examples of outlier detection applied on sub-trajectories are found in [77, 88, 140]. For outlier detection applied on parts of trajectories, each trajectory is first segmented into sub-trajectories and then some distance function or clustering method is used to detect outlying sub-trajectories. Finally, the trajectories to which the detected outlying sub-trajectories belong are identified as outliers. This approach has been followed in the TRAOD algorithm [77].

Outliers often appear as a by-product of other mining methods especially clustering. Because of this, most outlier detection approaches in the literature adopt some clustering algorithm and identify as outliers those objects that are not found in any cluster. Hence, the main issue is finding the appropriate distance measure that will discriminate these deviating trajectories.

A basic approach for detecting trajectory outliers analyzes the neighborhood of each trajectory by counting the number of neighbors or using a density-based clustering method. In this approach, the trajectories that have too few neighbors are categorized as outliers. The iBAT [146] anomaly detection framework follows an extended variant of this approach. In iBAT, the study area is partitioned into a grid structure and the trajectories crossing the same source-destination cell-pair are grouped. Finally, the approach uses an isolation mechanism to identify outlying trajectories exploiting the property that they are *few and different* in their groups.

Another category of approaches to trajectory outlier detection follow the procedures used in classification methods. For instance, a set of predefined features can be extracted from the trajectories, and then a standard distance measure applied on the extracted vectors. This approach was followed in [140] where distance measures have been applied on four features (direction, speed, angle, and location) to discover trajectory outliers. Alternatively, trajectory outliers can be discovered by training a two-label classifier model in which one label corresponds to normal trajectories while the other corresponds to abnormal trajectories according to the features under consideration. This approach was followed in [82].

3.2.3 Prediction

Prediction using trajectory data mainly aims at guessing the future location of a moving object based on existing trajectories. It has been especially motivated by the fast growing development of location-based services, one of its major application areas [50]. The objective of most of the studies on trajectory based prediction is to predict a location (destination or next location), but there are some studies which aim at predicting the entire route generally based on a road network. Trajectory based location prediction applies to a set of trajectories.

The studies on location prediction are based on two approaches (Markov models and sequential rules or Trajectory patterns) and can be classified into three categories: i) those considering only the data of the concerned moving object [50, 61, 71], ii) those considering only the data of other moving objects [9], and iii) hybrid approaches; i.e., considering both the data of the concerned object and those of other objects [104, 105, 137]. Studies based on Markov models use probabilistic models for location prediction. Examples of work following this approach can be found in [7, 98]. Studies based on trajectory patterns mine frequent patterns and association rules by defining a trajectory as an ordered sequence of time-stamped locations, and then using sequence mining methods. Examples of work following this approach can be found in [104,136,137]. The WhereNext approach [104] extracts T-patterns and builds a decision tree in which root-to-node paths are used for prediction. Ying et al. [136] use a similar approach extracting trajectory patterns to capture the movement behaviors that are motivated by geographic, temporal and semantic intentions from trajectories of users, and then matching the current movement of a user to the extracted patterns.

Unlike location prediction, which results in a location, the route prediction must produce a sequence of route segments to a location. The common approaches to route prediction fall in the following three categories:

- *Trip observations*: The approach involves observing several trips from a driver. For a new trip, the approach considers the start point and possibly a short starting section of the new trip, attempts to find a good match with a previous trip, and then uses the matched trip as a route prediction [41].

- *Markov model*: The approach builds a probabilistic model from the long-term trip history of the driver for predicting short-term route. The model looks at the driver's just-driven path and predicts the next road segment [69].

- *Turn proportions*: The approach builds a route by predicting which ways a driver will turn at successive intersections. For instance, Krumm [70] combines the information on which proportion of drivers will choose each turn option at an intersection and the driving times between pairs of road segments to make the turn prediction. This approach is based on the assumptions that drivers tend to choose the road that will give them more destination options and that they take the most efficient route in terms of time to their destination.

3.3 Relationships between trajectory mining methods

In this section we present the relationships between the different mining methods we have discussed. Table 1 presents these relationships showing primary methods and secondary methods as headings of rows and columns respectively. The secondary mining method at the top of a column uses the primary mining method on the left of a row to perform the task found in their intersection cell. For instance, the outlier detection method may apply clustering to group similar trajectories or sub-trajectories, which may be a required step for detecting trajectory outliers. In addition, an example of this relationship may be found in the reference entry [77]. The empty table cell means that we have not seen in the literature where pattern mining uses classification.

Primary methods	Secondary methods: Pattern mining	Secondary methods: Outlier detection	Secondary methods: Prediction
Clustering	• Grouping periodically related locations [20,85] • Grouping close trajectories [62,129] • Extraction of places of interest for frequent pattern mining [49] • Aggregating close locations for sequential pattern mining [45]	• Grouping similar trajectories or sub-trajectories [77]	• Grouping similar users [137] • Grouping similar trips of a user [3,41] • Grouping user's stay points for building trajectory patterns [136] • Grouping visited locations for building periodic patterns [61]
Classification		• Categorization into normal and abnormal trajectories [82]	• Categorizing on-going trajectory to one of defined trajectory clusters [3], or trajectory patterns [104]

Table 1: Relationships between trajectory mining methods.

4 Trajectory mining application problems

In this section, we discuss different application problems that have been or can be solved by mining trajectories. We categorize the application problems and relate them to generic trajectory mining methods discussed in previous sections. It is important to note that there is no one-to-one mapping between the methods and application problems. Different application problems may be addressed using the same method and one application problem may use more than one method; it depends on the specific tasks involved in addressing the application problem as shown in Table 2. The table relates application problems to the methods discussed in previous sections and provides references on where the application problem has been addressed using the methods mentioned.

4.1 Characterization of moving objects

The aim is to analyze the trajectories of moving objects for deriving some structure or information that describes the moving objects. Several instances of this application problem have been studied such as inferring activities of humans [117] or animals [120], determin-