IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 7, Issue 11, November 2018

# Survey of Trajectory Mining using Uncertain Sensor Data Model with Probabilistic Suffix Tree

**M.Gayathri[1], S.Nithyakalyani[2]**

PG Scholar, Department of Information Technology, K.S.R. College of Engineering, Tiruchengode, India[1]

Professor & Head, Department of Information Technology, K.S.R. College of Engineering, Tiruchengode, India[2]

**Abstract:** This paper describes a framework that works by collecting the trajectory data obtained from the sensors. The data is stored and processed in a way that helps in identifying events such as key activity areas, evolving activity, etc. It helps to attain better insight into the work habits of the population. Trajectory mining is either assumed that the time-ordered location data recorded as trajectories are either deterministic or that the uncertainty, e.g., due to equipment or technological limitations, is removed by incorporating some pre-processing routines. Thus, the trajectories are processed as deterministic paths of mobile object location data. Probabilistic trajectory extraction and mining from uncertain trajectory data is the first phase analysis on the subject. It is also interested in identifying and developing alternative approaches with the use of which can make the approach more scalable, e.g. a trajectory compression scheme could be developed to further decrease the length of the trajectories. This paper proposes an efficient distributed mining algorithm to jointly identify a group of sensor data and discover their trajectory of sensor data in wireless sensor networks. Then, Map-Reduce algorithm (Probabilistic Suffix Tree) is introduced which utilizes the discovered group trajectory sensor data shared by the transmitting node.

**Keywords:** Trajectory mining, Wireless Sensor Networks, Map-Reduce algorithm, Pattern mining

## I. INTRODUCTION

The Internet of things (IoT) is the network of physical devices, vehicles, home appliances, and other items embedded with electronics, software, sensors, actuators, and connectivity which enables these things to connect, collect and exchange data. IoT involves extending Internet connectivity beyond standard devices, such as desktops, laptops, smart phones and tablets, to any range of traditionally dumb or non-internet-enabled physical devices and everyday objects. Embedded with technology, these devices can communicate and interact over the Internet, and they can be remotely monitored and controlled. With the arrival of driverless vehicles, a branch of IoT, i.e. the Internet of Vehicles starts to gain more attention. Trajectory mining is an interesting data mining problem that has been studied in the context of smart cities and the Internet of Things (IoT). Smart cities and the IoT are indeed the way to the future as trillions of IoT devices, ranging from coffee machines to mobile objects which may or may not be inter-connected, generate enormous amounts of data which need to be modeled and processed effectively to improve daily life [5]. For example, to optimize the com-muting time to work, many sources of information including the intended route, calendar, city traffic, weather, etc. need to come together to determine a route which would be the most convenient and therefore, smart data collection, preparation and fast algorithms are needed which can work with the incoming data and propose solutions in real time. A trajectory is a time-ordered record of a moving object obtained at pre-defined discrete time intervals. However, the 'exact' location of a moving object during these intervals could be uncertain. A lot of research has focused on trajectory uncertainties with an aim to enhance the utility of trajectories. Probabilistic databases offer ways to model uncertainties using possible world semantics. [1]. The uncertainties in the trajectories could be at the event level, which is the uncertainty associated with the location of the object, or at the trajectory level, which is the uncertainty associated with the path recorded as compared to the path taken, or others. An interesting solution in this regard is to record the individual mobile object readings and then create complex events using probabilistic event extraction [3]. Many such systems have been proposed which work with trajectory.
The main objectives of the Trajectory Mining Using Uncertain Sensor Data are

- To consider uncertain sensor data and transform this to probabilistic trajectory data using pre-processing routines.
- To model this data as tuple level uncertain data.
- To propose dynamic programming-based algorithms to mine interesting trajectories.
- To propose a framework for probabilistic trajectory extraction and mining from uncertain trajectory data.

DOI 10.17148/IJARCCE.2018.71103

## II. RELATED WORKS

- **Yu Zheng [1]** in this article emphasized the importance of high- efficiency and reliable transmissions in VSNs for smart cities. Particularly, we study a case on traffic anomaly detection for VSNs by trajectory data analysis. Although VSNs can be regarded as the integration of social networks and IoVs to improve the quality of life for citizens, the avenues of VSN studies are not flat, and many open issues are still ahead. They believed that VSNs will draw extensive attentions and research efforts in the near future as the integrations of information technology and social network services become more compacted. This paper provides an overview on how to unlock the power of knowledge from trajectories, for researchers and professionals from no t only computer sciences but also a broader ranges of communities dealing with trajectories. At the end this paper, a list of public trajectory datasets have been given and a few future directions have been suggested.
- **Jie Bao and Tianfu He [2]** In this paper, they proposed a data driven approach to plan bike lanes based on the real bike trajectories collected from Mobike (a major station-less bike sharing system) in the City of Shanghai. The system can address the bike lanes planning problem in a more realistic way, considering the constraints and requirements from urban planners' perspective: 1) budget limitations, 2) construction convenience, and 3) bike lane utilization. They also proposed a flexible beneficial score function to adjust preferences between the number of covered users and the length of covered trips. The formulated problem is proven to be NP-hard, thus they proposed a greedy network expansion algorithm with two different initialization methods: top- k based and spatial clustering. Finally, in future work, they planned to use the parallel computing framework in Microsoft Azure to improve system response time to work more efficiently with larger trajectory datasets. Also, we would like to further explore the interactive planning process to incorporate more human intelligence.
- **Yanjie Fu and Yong Ge [3]** in this paper, they aimed to assess estate investment value by mining a variety of user-generated data. We collected a large scale of online user reviews and offline moving behaviors (taxi traces, smart card transactions, and checking) of mobile users. They indexed, filtered, propagated, distilled, aggregated mobile data, and extracted the fine-grained features from multiple perspectives (e.g., direction, volume, velocity, heterogeneity, popularity, topic, etc.) for evaluating estate values. However, since the extracted estate features usually are inter correlated and redundant, they proposed to teach a sparse pair wise ranker, which is mutually enhanced by simultaneously conducting feature selection and maximizing estate ranking accuracy. Finally, the experimental results with real world estate-related data demonstrate the competitive effectiveness of both extracted features and learning models.
- **Muhammad Muzammal and Rajeev Raman [6]** consider sequential pattern mining in situations where there is uncertainty about which source an event is associated with. They modeled this in the probabilistic database framework and consider the problem of enumerating all sequences whose expected support is sufficiently large. Unlike frequent item set mining in probabilistic databases, they used dynamic programming (DP) to compute the pr ob- ability that a source supports a sequence, and show that this suffices to compute the expected support of a sequential pattern. In classical SPM, the data to be mined is deterministic, but it is recognized that data obtained from a wide range of data sources is inherently uncertain. This paper is concerned with SPM in probabilistic databases, a popular framework for modeling uncertainty.
- **Prithu Banerjee and Sayan Ranu [8]** in this paper, they studied the problem of trajectory inference from partial observations. They developed a technique called InferTra that summarizes the entire possibilities through an "uncertain" trajectory. By taking the shape of an edge-weighted graph, an uncertain trajectory captures a richer representation of the uncertainty surrounding partial observations than maximum likelihood estimations. InferTra is built on the foundation of Gibbs sampling and is powered by a Network Mobility Model (NMM), which not only utilizes the spatial patterns embedded in historical data, but also unearths how these patterns vary with time. Extensive experiments on real network-constrained trajectories showed InferTra to be up to50% more accurate and20times faster than the state-of-the-art inferencing technique. In addition, an uncertain trajectory can handle a wider range of important queries.

## III. METHODOLOGY

### 3.1 Trajectory Data Extraction And Mining

In this section, we first discuss trajectory generation from the sensor data and then the trajectory mining.

**A. Trajectory Data:** A trajectory is a collection of location data points ordered by a time stamp. A data point is a triple of the form (eid, sid, e) where eid is the time stamp, sid is the source identifier, and e is the event. The length of a trajectory t is the number of data points it contains ∑|t|. For example, t = (a1, a2, a3), is a trajectory of three data points ordered in time. A trajectory s = (a1, a2 ••• an) is called a sub-trajectory of a trajectory, t = (b1, b2••• bn), if si = tj, for i,j ∈ [1,n],i ≤ j. In other words, we say that the trajectory t contains s. Given the location readings obtained from the sensors, a trajectory is obtained by combining in order all the location points recorded for a single object. The trajectory

DOI 10.17148/IJARCCE.2018.71103

database contains the trajectories for all the sources. The support of a trajectory t is the number of source trajectories that contain the trajectory t. The trajectory mining problem is defined as follows. Given a trajectory database D, find all interesting trajectories, i.e. trajectories whose support is at least a user-specified support threshold θ. Trajectory mining is a multi-stage process which primarily involves pre-processing and pattern mining. Trajectory data pre-processing is discussed first. variety of sources, e.g. sensors or other mobile devices, and is not entirely correct mainly due to equipment and technological limitations. The errors in trajectory data could be, e.g. (a) a location reading falling out of the motion track (or path) (b) or a moving object recorded at more than one distinct location, simultaneously. In all such situations, data has to be cleansed. For example, the value of the location attribute is fixed using techniques such as mean/median filters, etc.

**2) Data Compression:** Trajectory data is recorded at pre-defined discrete time intervals, e.g. a reading every few seconds by each sensor, and most of the points reported are usually repetitive and carry no significant information. However, keeping all the recorded data results in a notable increase in the computational complexity of the problem. It is a common practice to pre-process the data and reduce the number of exact locations recorded, i.e. attain some speed-up at the cost of losing some information which may not be worth the computation effort needed to process the information. Data compression could be offline or on the go and is performed using techniques such as computing the distance metric or similar.

**3) Trajectory Creation:** The final step is the trajectory creation which typically involves creating a trajectory of time-ordered location data points for each source. Once, the trajectories are created, data mining or other tasks could be performed. Another, important aspect is trajectory data management, however in this work we only focus on trajectory mining.

**B. Trajectory Mining:** Once the trajectories have been created, trajectory patterns are discovered which could be one of the following.

**1) Togetherness Patterns:** These patterns are aimed at answering questions such as which objects move together. This could help in identifying an emerging activity in a locality or similar. The local administration can use such information in better managing the city's resources, e.g. traffic signals.

**2) Common Path Patterns:** These patterns are the most frequent paths taken by the moving objects. The techniques used for finding such patterns include sequence mining, association mining, etc. These patterns generally help in predicting the next probable location of a moving object.

**3) Group Patterns:** Similar trajectories are grouped together to find groups of people who move similarly at the same points in time. This is not a trivial task as a feature vector has to be generated which is used to compute the distance between two trajectories. These group patterns show the group mobility trends and could be very useful when dealing with law and order situations, for instance.

**C. Uncertainty In Trajectories:** Uncertainty in trajectories is a major concern in many situations as the trajectory data recorded is only a sample of the actual movement. Further, the exact location of a moving object at a specific point in time may not be known. A lot of research has focused on working with trajectory uncertainties. An interesting aspect is to record the object locations along with the confidence values, i.e. along with the location, also record the confidence in an object being at that location. This is a novel idea as in literature the uncertainties associated to the object's location are removed using some threshold based methods and the final trajectory has only deterministic time-ordered data points. This work is focused on working with the uncertainty when dealing with trajectories. In the next section, we discuss the generation of probabilistic events from trajectory data and then present the probabilistic trajectory mining techniques to extract 'interesting' trajectories from the uncertain trajectory data.

### 3.2 Trajectory Mining Using Uncertain Events

Uncertain event extraction from the recorded sensor data is discussed. The data recorded by the sensors is simple location data and the accuracy of such data is usually low. Thus, the events extracted from such data are uncertain, as issue which is discussed below.

**A. Uncertain Events:** The most primitive type of events are presence events. A presence event is of the form (eid, sid, e, prob). For example, a sample reading looks like (t1, s2, l3, 0.7) which means that at time t1, a source s2 was spotted at location l3 with probability 0.7. The reason for having the probability is as follows. For example, a source s1 enters a room which has three (03) antennae installed to detect an object in the room. If two of the three antennae report the presence of the source in the room, there is only a 66.6%chancethatthesourcewas in the room at that time. It is also interesting that each antenna records the detection of an object with certainty, i.e. the reading from a single sensor looks

DOI 10.17148/IJARCCE.2018.71103

like (t1, s2, l3, 1.0) which means that the source s2 was sighted at location l3 at time t1 with probability 1.0. This makes sense as an antenna only reports an event which is detected and there is noun certainty in this simple event.

However, it is the readings from the neighboring antennae which contribute to the belief in the presence of a source at a specific location. The system takes the readings from multiple antennae and only then decides the confidence in a presence event. In a subsequent formulation, suppose a source s1 also enters the room and one out of the three antennae detect s1. What is the probability of the event that both s1 and s2 are in the room, together? An event of this form could be that the sources s1 and s2 are at location l3 with probability 0.18.

However, the detection of the sources s1 and s2 at locationl3 maynotbesufficienttoestablishthatboths1 ands2 are at location l3. There is other information which should also be considered whilst creating such events, e.g. the ownership of the location l3. If one of the two sources is the owner of this location, it is probable that the two are together, for example for a meeting. However, if neither of the two owns the location, the detection of these two at the same place with low confidence, i.e. probability 0.18, could be an error. Therefore, the detection and other information are also needed to establish such complex events and for establishing the truth value of a true occurrence. Further, the sensors are not entirely accurate due to technological limitations, and for example, if a sensor has an error rate of 20% and there are a total of three (03) sensors at a point, then a presence event has a low accuracy.

**B. Uncertain Data Model:** From the previous section, we know that the uncertain events generated by the sensors are of the form (eid, sid, e, prob) which corresponds to tuple-level uncertainty, i.e. a tuple has an existential probability of occurrence. A sample probabilistic trajectory database is shown in Table 1. We now define the possible world semantics for an uncertain trajectory database D0.

Table 1. A sample probabilistic trajectory database.

| **Time-Stamp** | **Sid** | **Eid** | **Probability** |
|---|---|---|---|
| 1 | T1 | U | 0.4 |
| 2 | T2 | W | 0.6 |
| 3 | T2 | V | 0.7 |
| 4 | T1 | V | 0.8 |

1) **Possible Worlds Semantics**: The possible world semantics are as follows. Given an uncertain trajectory database D0, for each event e in a trajectory there are two kinds of worlds: one in which the event is present and the other where it is not. For each source trajectory ti, the set of possible worlds is obtained by taking all possible combinations in which an event is present in the world or otherwise. The complete set of possible worlds is obtained by taking all such combinations. The probability of an event occurring is the cumulative probability of occurrence of the worlds where this event is present. As in the literature, we assume that the events across possible worlds occur independently of each other. An example of possible world computation is shown in Tables 2-4 for the sample database of Table 1 transformed to a trajectory database in Table 2.

Table 2. The trajectory database of table 1 transformed to probabilistic trajectories.

| **Trajectory Id** | **Trajectory** | **Trajectory Id** |
|---|---|---|
| t1 | (u:0.4, v:0.8) | t1 |
| t2 | (w:0.6, v:0.7) | t2 |

Table 3. The set of possible worlds for source t1 from table 2.

| **World** | **Probability** |
|---|---|
| T1,1 | <>= (1-0.4) * (1-0.8) = 0.12 |
| T1,2 | {u} = 0.4 * (1 – 0.8) = 0.08 |
| T1,3 | {v} = (1-0.4) * 0.08 = 0.48 |
| T1,4 | {u, v} = ).4 * 0.8 = 0.32 |

Table 4. Complete set of possible worlds for the trajectory database of table 2.

| **World** | **T1** | **T2** | **Pr(D'*)** |
|---|---|---|---|
| D'1 | <>= 0.12 | <>= 0.12 | = 0.12 * 0.12 |
| D'2 | <>= 0.08 | {v}= 0.08 | = 0.08 * 0.18 |
| … | … | … | … |
| D'4 | {u,v } = 0.32 | {w,v } = 0.12 | = 0.32 * 0.42 |

2) **Interestingness Measure:** Using the possible world semantics, an event that occurs in a significant number of worlds with high probability is considered an interesting event. The interestingness measure, the expected support of an event, is defined in terms of the expectation of the event occurring in all the possible worlds, i.e. for a trajectory t, For

example, the expected support of a trajectory {u, v} in the sample database of Table 1 is computed by taking the sum of the probabilities of all the possible worlds which contain {u, v}, i.e. worlds D'12−D'16, as shown in Table 4.

$$ExpSup\left(t, D'\right) = \sum\nolimits_{D^* \in PW(D')} \Pr\left[D^*\right] * Sup(t, D^*).$$

**3) Uncertain Pattern Mining:** The uncertain pattern mining problem is defined as follows. Given a trajectory database, find all frequent patterns whose expected support is at least a user-defined support threshold θ. Note that the number of possible worlds is exponential in nature and computing the expected support using possible worlds becomes computationally intractable. We now present a dynamic programming approach to compute the expected support of a trajectory.

### 3.3 Expected Support Computation

Given a trajectory and a source trajectory, we create a dynamic programming matrix M,(q+1)×(p+1), where q is the number of elements in the trajectory and p is the number of elements in the source trajectory, and initialize all elements in the top row equal to 1 and all elements in the first column (except the top entry) equal to 0. Next, we compute the other values row-by-row by using the following relation:

$$\mathbf{M[i, j] =(1cij)\times M[i,j-1]+cij \times M[i-1,j-1]}$$

An example of this computation is shown in Table 5. The right bottom cell in the table gives the expected support of the trajectory {u, v}.

Table 5. Computing expected support using dynamic programming

| | | **{u:0.4}** | **{v:0.8}** |
|---|---|---|---|
| | | **1** | **1** |
| **{u}** | **0** | **0.4 * 1 + ( 1 – 0.4) * 1 = 0.4** | **0.4** |
| **{u, v}** | **0** | **0** | **0.4 * 0.8 + ( 1 – 0.8) * 0 = 0.32** |

The expected support of a trajectory t in the trajectory database D0 is computed by summing the expected support of t across all trajectories.

**Algorithm 1:** An Outline of the Trajectory Mining Algorithm

```
Given: A Trajectory Database D', An Expected Support threshold □
Required: All frequent trajectories
i=2
F1 : Compute all simple events
while Fi−1 is not null
Ci =: join Fi−1 with itself
Prune Ci
for all trajectories in Ci
Compute Expected Support
end for
Fi =: all frequent Ci
i=: i+1
end while
Output the frequent trajectories in F
```

**3.4 Uncertain Trajectory Mining:** The uncertain trajectory mining algorithm is given in Algorithm 1.

**1) Frequent Simple Events:** A scan of the trajectory database D' extracts all simple events in the database D' which have support at least equal to the threshold. The expected support of all the events is computed. Once the database has been scanned, all the frequent simple events have been found and these form the candidates for the next phase, i.e. frequent pair computation.

**2) Frequent Pairs:** Once the frequent simple events have been computed, all possible pairs of events are generated which are then tested for being frequent. Note that only the frequent simple events are used to generate candidate frequent pairs. This is due to the Apriori property which is anti-monotonic and states that for any pair event to be frequent, both the simple events in the pair have to be frequent. For example, for {u, v} to be frequent, both {u} and {v} need to be frequent. Only then should the support computation test be performed.

**3) Frequent Trajectories:** The frequent pairs discovered during the previous phase are used to generate candidate trajectories by appending the frequent simple events to the frequent pairs. The idea is that a candidate trajectory can be extended by appending a simple event to a frequent trajectory which has already been discovered. This step continues until no more candidate trajectories can be created or all the frequent trajectories have been discovered.

## CONCLUSION

In this paper, proposed a framework for probabilistic trajectory extraction and mining from uncertain trajectory data. This is the first study on the subject and many interesting directions need to be explored, e.g. going beyond the number of sources and hours considered in this study. It shows interest in identifying and developing alternative approaches with the use of which we can make the approach more scalable, e.g. a trajectory compression scheme could be developed to further decrease the length of the trajectories. Further, an approximation scheme could be developed to avoid the dynamic programming processing at the cost of some accuracy. The inherent independence of the trajectories could be used to adapt the proposed algorithm to a distributed computing environment, e.g. Map-Reduce. This work has focused on mining uncertain trajectories. An insight into the discovered trajectories and assessment of the usefulness of the trajectories could also be subjects for interesting future work. The work uses expected support as the interestingness measure. Other interestingness measures like probabilistic suffix tree based mining should be the future work. Also data compression and decompression so that minimum data can be send to base station (data collecting) node is to be carried out.

## REFERENCES

[1]. Y. Zheng, ''Trajectory data mining: An overview,'' ACM Trans. Intell. Syst. Technol., vol. 6, no. 3, p. 29, 2015.

[2]. J. Bao, T. He, S. Ruan, Y. Li, and Y. Zheng, ''Planning bike lanes based on sharing-bikes' trajectories,'' in Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Aug. 2017, pp. 1377–1386.

[3]. Y. Fu et al., ''Sparse real estate ranking with online user reviews and offline moving behaviors,'' in Proc. IEEE Int. Conf. Data Mining (ICDM), Dec. 2014, pp. 120–129.

[4]. B. Fazzinga, S. Flesca, F. Furfaro, and F. Parisi, ''Cleaning trajectory data of RFID-monitored objects through conditioning under integrity constraints,'' in Proc. 17th Int. Conf. Extending Database Technol. (EDBT), Athens, Greece, Mar. 2014, pp. 379–390.

[5]. P. Wang, Y. Fu, G. Liu, W. Hu, and C. Aggarwal, ''Human mobility synchronization and trip purpose detection with mixture of Hawkes processes,'' in Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Aug. 2017, pp. 495–503.

[6]. M. Muzammal and R. Raman, ''Mining sequential patterns from probabilistic databases,'' Knowl. Inf. Syst., vol. 44, no. 2, pp. 325–358, 2015.

[7]. Y. Li et al., ''Sampling big trajectory data,'' in Proc. 24th ACM Int. Conf. Inf. Knowl. Manage. (CIKM), Melbourne, VIC, Australia, Oct. 2015, pp. 941–950.

[8]. P. Banerjee, S. Ranu, and S. Raghavan, ''Inferring uncertain trajectories from partial observations,'' in Proc. IEEE Int. Conf. Data Mining (ICDM), Shenzhen, China, Dec. 2014, pp. 30–39.

[9]. M. Li, A. Ahmed, and A. J. Smola, ''Inferring movement trajectories from GPS snippets,'' in Proc. 8th ACM Int. Conf. Web Search Data Mining (WSDM), Shanghai, China, Feb. 2015, pp. 325–334.

[10]. Z. Feng and Y. Zhu, ''A survey on trajectory data mining: Techniques and applications,'' IEEE

[11]. [1] CIR-11, 2014, pp. 8–23.

DOI 10.17148/IJARCCE.2018.71103