1K1-IOS-1a-1

# A Study on Next Location Predictive Modeling using Mined Temporal Sequential Patterns as input to a Decision Tree

Danaipat Sodkomkham, Roberto Legaspi, Satoshi Kurihara, Masayuki Numao

The Institute of Scientific and Industrial Research, Osaka University

Knowledge about position of the participants is commonly used in location-based services and applications in smart environment, which need to know an approximated location of the users to provide a proper service. Furthermore, when users are moving in an environment doing tasks, knowledge of the next location or destination of those movements can be used to assist the system to give more accurate system responses. These services require the following knowledge to operate: 1) a predicted location of the users or a plausible destination, and 2) a predicted time of arrival. For these two requirements, a predictive temporal sequential pattern mining algorithm is proposed in this paper, which is a method aimed at predicting the next location of a moving object from its temporal and spatial context. The prediction uses previously extracted temporal sequential patterns, which represent behaviors of moving objects as sequences of locations frequently visited within a certain speed. A decision tree based classifier is trained from the temporal sequential patterns and used as a predictor for the next location that is most probable location to be visited within the movement sequence. Finally, a performance evaluation of the method and over a real dataset is presented.

## 1. Introduction

In the design of a smart environment or intelligent space, knowledge about human behavioral patterns is important when designing services, and applications that could capture user patterns, predict their needs, and provide a proper system response are desirable. Particularly, a model of user mobility pattern in the space is required by various kinds of services. Hence, we focus on developing a system that can learn user movement patterns and make a prediction about the future location, where a user is heading and the time of his/her arrival to that location.

The proposed method in this paper is composed of two parts. First, the temporal sequential pattern mining algorithm is specially designed to extract the movement patterns that occur at a certain frequency. The movement patterns are represented by a sequence of locations and time interval between two locations in a sequence. This time interval factor is used to indicate the movement speed of an object. Second, the predictive method uses previously extracted movement patterns to train its classifier to predict the next user location.

The three main steps of our approach are as follows:

**Movement Detection:** In our experiment we use the infrared (IR) sensors that are basically used for distance measurement to detect a state, where there an object is at a certain location in the experimental space. Afterwards, sequences of movements are generated by concatenating these events together. A concise representation of these movement sequences consists of two components: 1) a location id, and 2) timestamp of the visit. Finally, each occurring sequence for each movement is logged into the sequence database. Note that our sensors itself cannot distinguish different people. Furthermore, if two or more users move at the same moment, those movements will be detected as one mixed movement sequence. However, IR sensors are unobtrusive, i.e. they need not be attached on subjects and permit the subjects to move more naturally in the space. Because of this advantage, we prefer the IR sensors to cameras or RFID tags.

**Temporal Sequential pattern Extraction:** From the sequence database acquired previously, the temporal sequential pattern mining algorithm is executed to extract frequent patterns of movements with a typical movement speed. The method we use in this step is modified from the well-known PrefixSpan algorithm [7] to support temporal context that indicates movement speed. The algorithm is explained more in detail in section 3.

**Predictive model construction:** Given the temporal sequential patterns, we trained a decision tree-based classifier, called C4.5 [14], to identify to which in a set of 'next location' classes will any new observed movement sequence belong. To test the classifier, given a new movement sequence *S*, we used the prediction tree to predict the next location of *S*.

The performance of the method is evaluated against a real dataset over 5 weekdays of movement sequences collected by the IR sensors network installed on workspaces, hallway, and tearoom in our laboratory. An experiment is designed to evaluate the prediction accuracy of the method. We use the predictor trained from previous 24 hours of movement patterns to predict the next location of a new movement in real time to evaluate the prediction accuracy. The results of the experiment show that considering temporal context helps us achieve higher accuracy and efficiency.

The rest of the paper is organized as follows. Section 2 reviews related works. Section 3 describes the temporal sequential pattern extraction in detail. Section 4 explains how to build a predictive model from sequential patterns extracted from the previous step, and discusses how to predict the next location for a newly observed movement using the predictive model. The experimental results and performance study are presented in section 5, and we conclude our study in section 6.

Contact: Danaipat Sodkomkham, ISIR, 8-1 Mihogaoka, Ibaraki, Osaka, 567-0047, Japan, +81-6-6879-8426, +81-6-6879-8428, danaipat@ai.sanken.osaka-u.ac.jp

## 2. Related work

In this section we summarize some relevant studies related to the sequential patterns mining and location prediction.

### 2.1 Sequential Pattern Mining

Sequential pattern mining algorithms are designed to extract a set of frequently observed orders or subsequences as patterns from a dataset called sequence database. The sequence database consists of ordered events or items that are annotated with or without time. There are many researchers who study on sequential pattern mining [1, 5, 7, 9]. Since its performance is the best compared to GSP [5] and SPADE [9], the PrefixSpan [7] algorithm has been applied to many problem areas and has been provided with various extensions and modifications in different directions. However, sequential patterns extracted from PrefixSpan do not provide knowledge about the time span between items in the sequential patterns that could further support decision making. One solution to this problem has been introduced in [2], that is time-interval sequential mining algorithm was designed to extract not only frequently occurring sequences but also the time intervals between successive items. Time intervals have been commonly presented as having definite range [3, 7]. This, however, has led to strict or rigid boundary problems. When a time interval is near the boundary of two adjacent predetermined ranges, the tendency is to either ignore or overemphasize the interval [8]. Hence, in our previous study [13] we modified the Apriori algorithm to deal with the sharp boundary problem given multiple time intervals using clustering wherein we can specify a number of clusters. We used in particular the k-means method. However, its performance is not good enough to create a promising predictive model, which becomes the target of this work. Hence, we would like to modify to better predict the future location of a moving human subject. The proposed method in this paper also uses a concept of cluster analysis that deal with multiple time interval values with clusters of time intervals that are similar to each other within the same cluster as opposed with to those in other clusters. We integrate this idea into the PrefixSpan algorithm to create a temporal sequential pattern mining algorithm that allows multiple time interval sequential patterns with more flexible time interval data.

### 2.2 Trajectory Pattern Mining and Location Predictor

The concept of clustering sequential patterns and time intervals was also discussed in [11]. A trajectory pattern mining algorithm was a PrefixSpan based sequential pattern mining algorithm specially designed for spatio-temporal datasets. The algorithm finds a set of sequences of regions that are frequently visited by subjects and with typical time intervals. In [11], the time intervals are clustered using a density based clustering algorithm to handle multiple-time intervals between regions. Consequently, the sequential patterns found from the trajectory pattern mining algorithm are used to build a predictive model called “WhereNext” [12]. WhereNext is a location predictor that uses trajectory patterns as predictive rules. The data structure called the prediction tree [12], is constructed using all sequential patterns. The nodes of the tree are regions and edges representing represent typical time intervals between two successive regions. A matching method needs to be defined afterwards to match test instances with the model and make a prediction.

The difference between the work of WhereNext and ours is that we use a decision tree learning algorithm to build a prediction tree instead of manually selecting attributes for every node and edge of the tree without knowing how well each attribute separates the training examples according to their target class (i.e., the region where the subject is moving towards). This obviously makes our prediction tree smaller and predict faster. However, the time complexity for constructing the prediction is much higher in our case. We use a decision tree learning algorithm called C4.5, which has a time complexity of $O(mn^2)$, where $m$ is the size of the training examples and $n$ is the number of attributes. On the other hand, the time complexity of the prediction tree construction phase in WhereNext is $O(lm)$, where $l$ is an average length of the patterns and $m$ is the size of the training examples.

## 3. Temporal Sequential Pattern Mining

In this section we propose the Temporal Sequential Pattern Mining (TSPM), an algorithm that modifies PrefixSpan to determine temporal context in the candidates pruning step. The PrefixSpan algorithm adopts a pattern-growth approach to sequential pattern mining as developed by Pei et al. [7]. PrefixSpan is a divide-and-conquer algorithm that extracts subsequences that appear in a dataset with frequency no less than a user-specified threshold. The first scan finds length-1 sequential pattern that satisfy the minimum threshold. Each sequential pattern is treated as a prefix and used to project over the dataset to find longer sequential patterns. This process recurs until there is no more prefix left to be projected. The summary of the PrefixSpan can be found in Algorithm 1.

Basically, our movement sequences have temporal annotations that imply speeds. For example, (A, 10:20am)→(B, 10:22am)→(C, 10:30am) can be read as “a person takes 2 minutes from location labeled ‘A’ to location ‘B’ and 8 minutes from ‘B’ to ‘C’. The main idea of our approach that we add into PrefixSpan is that if we could extract typical movement speeds of particular movement patterns, the algorithm can ignore all projected candidate sequences that happened in infrequent speeds. An example of such cases is when it is 10 kilometers between ‘B’ and ‘C’. It is obviously impossible for a person to walk from ‘B’ to ‘C’ within 10 minutes, which is also uncommon time interval found in movements between ‘B’ and ‘C’. Hence, the algorithm can reject this sequence pattern because it is not practical.

TSPM uses a clustering algorithm called k-means clustering [15] to assign in groups time intervals appearing in each of the sequences. The clustering step is put into the PrefixSpan in frequency checking step. The projected sequential patterns that have lower frequency than the user-specified threshold, i.e., minimum support, will be discarded and the patterns that were clustered into a group that has smaller size than a minimum threshold, i.e., the minimum t-support, will be discarded as well. The pseudo-code of TSPM is presented in Algorithm 2.

Algorithm 1: PrefixSpan
**Input:** A dataset *seqDB* of sequences database, and a minimum support *minSupp*
**Output:** A set of sequential patterns

```
L=1;
Prefix_{L=1} = findLengthL1patterns(seqDB)
while Prefix_L ≠ 0 do
    Prefix_{L+1} = {}
    for each Prefix in Prefix_L do
        if Prefix.support > minSupp do
            output(Prefix)
            ProjectedSequence = project(seqDB, Prefix)
            Prefix_{L+1}.add(ProjectedSequence)
        end if
    end for
    L++
end while
```

Algorithm 2: Temporal Sequential patterns Mining (TSPM)
**Input:** A dataset *seqDB* of sequences database, and a minimum support *minSupp,* a minimum temporal support *minTSupp*, and number of time interval clusters *k* for k-means clustering algorithm.
**Output:** A set of temporal sequential patterns in a set of couples (sequence pattern, time interval cluster)

```
L=1;
Prefix_{L=1} = findLengthL1patterns(seqDB)
while Prefix_L ≠ 0 do
    Prefix_{L+1} = {}
    for each Prefix in Prefix_L do
    if Prefix.support > minSupp do
        timeintervalClusters = k-means(Prefix.timeinterval, k)
        for each cluster in timeintervalClusters do
            if cluster.size > minTSupp do
                output(Prefix, cluster)
                ProjectedSequence = project(seqDB, Prefix)
                Prefix_{L+1}.add(ProjectedSequence)
            end if
        end for
    end if
  end for
  L++
end while
```

## 4. Predictive Model Building

Our approach uses a decision tree-based classifier to identify to which of a set of 'next location' classes will a newly observed movement sequence belong. All sequential patterns extracted from TSPM algorithm are used as training examples for the classifier. Based on our hypothesis that temporal context will help the predictor achieve more accuracy, the time intervals and timestamps of each of sequence are also included as classifier features. The time interval attributes are time durations that subjects usually take from one location to another, and the timestamps are basically time logs that indicate frequent periods (in hour unit of time) of the day that these patterns were observed frequently. Consequently, features of the model include 1) length *n*-1 sequences of locations, when *n* is the length of the patterns, time intervals, and timestamps. Finally, the last element of the sequences is treated as a target class that we want the classifier to identify.

Given a prediction tree previously built *T* and a new movement sequence *S*, *T* classifies *S* to a target class that has the highest possibility to be the next location of *S.* Classification uses previously visited locations, time intervals between successive locations in the sequence, and an observed timestamp of the sequence for its decision.

## 5. Experiment Results

We have conducted a performance study to evaluate the accuracy of the proposed method and compared this with the predictive model built by PrefixSpan without any temporal context. As in [13], dataset from IR sensors are used. 60 IR sensors have been installed extensively in our experimental space as shows in Figures 1, 2 and 3. The experimental spaces are composed of three main sections. First, the student room, where we installed a total of 36 sensors in student workspace cubicles to detect movement from students. Second, we installed 20 sensors in the tearoom, to detect usage of the printer (and copy machine), teapot, refrigerator, microwave oven, kitchen sink, couches, and TV. This space is shared among students and all laboratory members. Sensors were installed to detect movements and also to know the usage of facilities provided in the room. Finally, we installed 4 sensors along the hallway, between the student room and the tearoom to observe movements between these two sections. We collected mobility data of 5 weekdays.

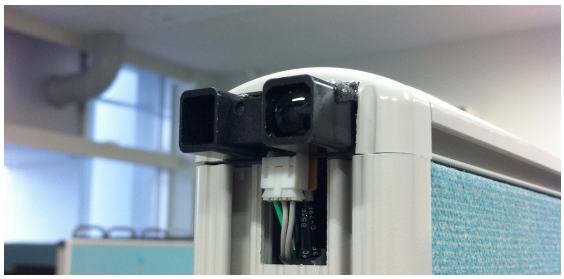

Figure 1: IR Sensor setup


Figure 2: IR sensors installed in the student room

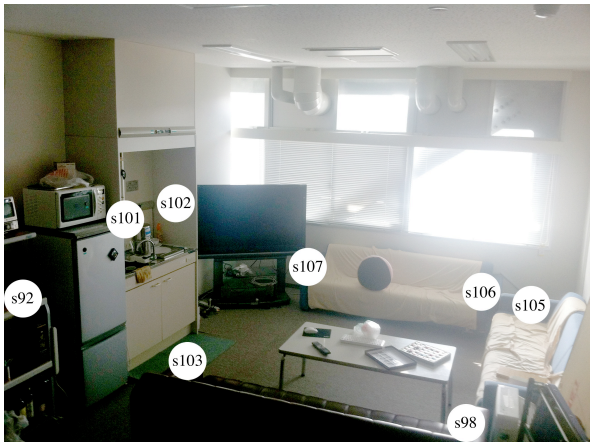


Figure 3: IR sensors installed in the tearoom

All experiments were performed on a 2.67GHz Intel Xeon PC with 6GB of main memory. All the algorithms are implemented in Java. The parameters of TSPM are set as follows. The minimum support is set to 10 occurrences/day and the minimum t-support is 2 data points/day. Specifically, the minimum support used in this setting is an absolute support value, which is basically a frequency. This means that we concern ourselves with only movements that happened at least 10 times in one day. In the same way, the minimum t-support indicates the size of a time interval cluster. This simply means that a certain path must be traversed with the similar speed at least twice. Similarity in this case is determined by the time interval cluster, where each sequential pattern was assigned to during the clustering step. Lastly, parameter *k* in the k-means is set to 5.

Table 1 compares the precision of two approaches. First, the proposed method, which is concerned with the temporal context of the movement patterns, and the second approach that ignores all temporal contexts. A 10-fold cross validation was used for every dataset. The result clearly shows that temporal context helps the model predict more accurately.

Figure 4 shows the performance of TSPM and PrefixSpan in detail. F-measure is used to evaluate the prediction accuracy. The 5-day dataset is used to train and test the model using 10-fold cross validation. The column labeled as class is the next location that we want the algorithm to predict. For example, s92 is a sensor ID representing an area around a teapot; s95 is at the door, and s35 is at a student's desk. The results show the accuracy of two approaches using the F-measure. We use F1-score for the F-measure. The performance of TSPM is higher than the PrefixSpan for all classes and shows no problem with the skewed dataset. Unlike the PrefixSpan without temporal context approach, it performs poorly especially in s13, s26, s34, and s103, which have nearly zero in both precision and recall rate (See Figure 5 for the number of training examples per class).

Table 2 shows the accuracy of the two approaches in practical application. Two different datasets were used for training predictive model and for testing. The prediction tree is built from the temporal sequential patterns extracted from a dataset collected one day before, and the dataset collected in the next day is used for testing. We measured the accuracy of two approaches by a ratio of correctly predicted location and incorrectly predicted location. The TSPM approach also gives slightly better accuracy than the PrefixSpan. However, a low performance in both approaches on Thursday indicates that a predictive model built on a small dataset on Wednesday cannot produce a representative model for most of the movement that happened on Thursday. One plausible solution for this problem is increasing the size of the datasets.

| *Dataset* | *Mon* | *Tue* | *Wed* | *Thu* | *Fri* | *All5days* |
|---|---|---|---|---|---|---|
| TSPM | 93.2% | 88.7% | 91.0% | 97.0% | 90.4% | 95.7% |
| PrefixSpan | 93.2% | 75.4% | 63.3% | 69.8% | 63.3% | 68.9% |
| Size of the dataset | 2,965 | 2,399 | 4,217 | 46,103 | 1,631 | 57,315 |

Table 1: Precision of TSPM and PrefixSpan w/o temporal context

| *Training* | *Mon* | *Tue* | *Wed* | *Thu* |
|---|---|---|---|---|
| *Prediction* | *Tue* | *Wed* | *Thu* | *Fri* |
| TSPM | 78.1% | 49.4% | 23.1% | 58.0% |
| PrefixSpan | 78.1% | 44.9% | 22.1% | 48.3% |

Table 2: Accuracy comparison between TSPM and PrefixSpan in a Next Day prediction experiment

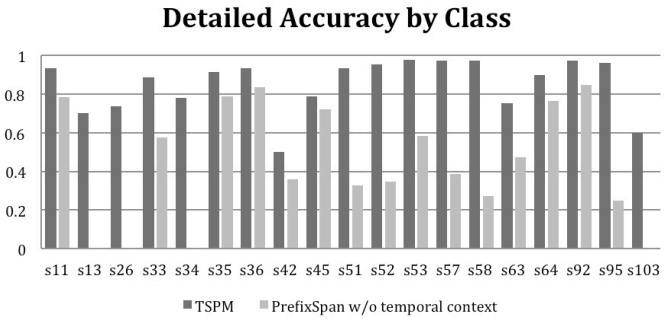


Figure 4: Detailed Accuracy by Class measured by F-measure (F1-score)

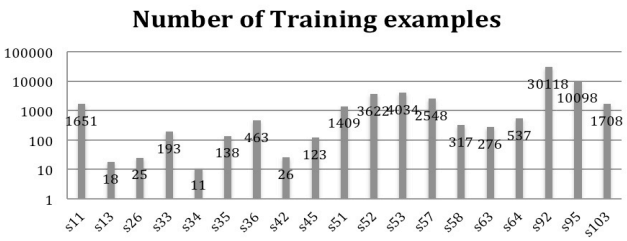


Figure 5: Size of a training set per class

## 6. Conclusion

To design a service or application in a smart environment research, we need to model human movement behavioral patterns where the knowledge of future user movement is vital. We designed a sequential patterns mining algorithm, called Temporal Sequential patterns Mining algorithm (TSPM) especially for analyzing human movement patterns. Afterwards, the movement patterns are used to build a predictive model that was constructed by C4.5. A future location is treated as a class where a newly observed movement sequence will be classified. One hypothesis that we made in this work is that temporal context of such movement patterns (e.g., time span between two locations, typical time of the day that a certain pattern appears) could help the predictor perform with smaller error. TSPM is designed to handle multiple time intervals that appear in sequential patterns by employing k-means clustering, a specific number of clusters *k* can explicitly stated. Different from [13], we developed TSPM based on the PrefixSpan algorithm as an alternative to an Apriori algorithm because of its speed and less memory complexity problems. Lastly, the prediction part of our approach is different from [12] since we used decision tree learning to build the prediction tree, which basically greedy selects an attribute that is most useful for classifying examples to create a decision node first. This gives a more compact and smaller tree than a prediction tree directly constructed by the sequential patterns.

We implemented and tested our approach against the prediction method that do not use temporal contexts. The results show that the temporal context of movement speeds and time helps the predictor to achieve higher performance by 27% in the average.

We also conducted an experiment that simulated a real application, where movement patterns are mined and the predictive model is built beforehand offline, and used to predict the future location for a movement in the succeeding day. The result also shows an improvement in prediction accuracy by 2% on the average, whereas the test cases with lowest and highest improvement got 0% and 10% respectively. The reason behind this situation is that the number of users cannot be fixed; therefore we got a different number of movement patterns on different training days. Too small number of training examples could lead to problems because they cannot be used to generate a representative model for all cases.

### Acknowledgements

This work is supported in part by the Management Expenses Grants for National Universities Corporations through the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan, by the Global COE (Centers of Excellence) Program of MEXT and by KAKENHI 23300059.

### References

1. Agrawal, R., Srikant, R.: Mining sequential patterns. In: Proc. 11th International Conference on Data Engineering. pp. 3–14 (1995)
2. Chen, Y.L., Chiang, M.C., Ko, M.T.: Discovering time-interval sequential patterns in sequence databases. Expert Systems with Applications 25, 343–354 (2003)
3. Chen, Y.L., Huang, T.C.K.: Discovering fuzzy time-interval sequential patterns in sequence databases. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics 35(5), 959–972 (October 2005)
4. Hafez, A.: Association of dependency between time series. In: Proc. SPIE vol. 4384, SPIE Aerosense (2001)
5. Hirate, Y., Yamana, H.: Generalized sequential pattern mining with item intervals. Journal of Computers 1(3), 51–60 (June 2006)
6. Hu, Y.H., Hang, T.C.K., Yang, H.R., Chen, Y.L.: On mining multi-time-interval sequential patterns. Data and Knowledge Engineering 68, 1112–1127 (2009)
7. Pei, J., Han, J., Mortazavi-Asl, B., Wang, J., Pinto, H., Chen, Q., Dayal, U., Hsu, M.C.: Mining sequential patterns by pattern-growth: The PrefixSpan ap- proach. IEEE Transactions on Knowledge and Data Engineering 16(11), 1424–1440 (November 2004)
8. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalization and per- formance improvements. In: Proc. 5th International Conference on Extending Database Technology: Advances in Database Technology (1996)
9. Zaki, M.J.: Spade: An efficient algorithm for mining frequent sequences. Machine Learning 42(1-2), 31–60 (January-February 2004)
10. Jiawei H, Hong C, Dong X, Xifeng Y., Frequent pattern mining: current status and future directions. Data Mining and Knowledge Discovery, Volume 15 Issue 1, August 2007
11. F. Giannotti, M. Nanni, F. Pinelli, and D. Pedreschi. Trajectory pattern mining. KDD 2007: 330-339.
12. A.Monreale , F. Pinelli , R. Trasarti , F. Giannotti, WhereNext: a location predictor on trajectory pattern mining, Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, June 28-July 01, 2009, Paris, France.
13. R. Legaspi, D. Sodkomkham, K. Maruo, K. Fukui, K. Moriyama, S. Kurihara, and M. Numao (2012) Time-Interval Clustering in Sequence Pattern Recognition as Tool for Behavior Modeling. In Theory and Practice of Computation - Proceedings in Information and Communications Technology, 5. (To Appear)
14. Quinlan, J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993.
15. MacQueen, J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations. 1. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press. pp. 281–297. MR0214227. Zbl 0214.46201. Retrieved 2009-04-07.