Comparative mining and clustering of temporal route visit profiles

Can Yang and Győző Gidófalvi

KTH, Royal Institute of Technology in Sweden
Email: {cyang, gyozo}@kth.se

1. Introduction

Large collections of trajectory data is being generated nowadays, which provide data-driven solutions to travel time estimation, traffic flow prediction or travel behaviour modelling. These solutions often discover and utilize various patterns in trajectory data in form of point clusters (Li *et al.*, 2012), origin and destination (OD) matrices (Toole *et al.*, 2015) or sequential patterns (Giannotti *et al.*, 2007; Ye *et al.*, 2009; Zheng, 2015). A *sequential pattern* in a trajectory set is defined as a common sequence of locations visited by a certain number of trajectories (Zheng, 2015). *Contiguous sequential pattern* (CSP) further requires the locations to be *spatially contiguous*, such as adjacent road edges in a route (Yang and Gidófalvi, 2018b). Therefore, a CSP in a trajectory set provides the visiting frequency of a specific route. An example is illustrated in Figure 1 where the elements in a sequence represent the ID of a road edge. Let $\langle i_1, i_2, \cdots, i_n \rangle : s$ denote a sequential pattern $\langle i_1, i_2, \cdots, i_n \rangle$ with support or frequency of s. In Figure 1, $\langle 1, 3 \rangle$ is a sequential pattern but not a CSP as the elements 1 and 3 are not contiguous in the sequence set. Different from traffic flow which counts the vehicles passing an individual edge, CSP covers a sequence of edges, which is valuable in identifying long-term regularities in the movements such as route choice between specific ODs and traffic flow at intersections.

In practice, when CSP are mined from map matched trajectories, a closedness constraint is commonly added because the original CSP set generally contains a lot of redundancy. As shown in Figure 1, the CSP $\langle 1, 2, 3, 4 \rangle : 3$ implies the existence of $\langle 1 \rangle : 3$,$\langle 1, 2 \rangle : 3$ and $\langle 1, 2, 3 \rangle : 3$. By adding the closedness constraint, CSP that are contiguously contained by another CSP with the same support are pruned as they can be losslessly reconstructed. The mining result is called a closed CSP (CCSP) set, which is substantially smaller than the original CSP set.

Yang and Gidófalvi propose approaches to efficiently extract CCSPs from a large collection of trajectories in two steps: map matching (Yang and Gidófalvi, 2018a) and CCSP mining (Yang and Gidófalvi, 2018b). In their work, the change of route visit frequencies over time can only be visually inspected by comparing maps in various time periods. Quantitative information about the change is generally unavailable in visual comparison and it is difficult to compare more than two periods, e.g., the change of frequency of a route in 24 hours.

To extract quantitative information of the change of route visit frequencies, this paper proposes a novel *comparative CCSP mining* algorithm which takes multiple input CCSP sets as input and exports the frequency distribution of CCSPs

Sequence set

Sequence
1,2,3,4
1,2,3,4
1,2,3,4,5
2,3,4,5
6,2,3,4,5
6,2,3,4

Sequential pattern set (min_sup = 2)

Sequential pattern	Support
1	3
1,2	3
1,2,3	3
1,2,3,4	3
1,3	3
1,3,4	3
…	

Contiguous sequential pattern set (min_sup = 2)

Contiguous sequential pattern	Support
1	3
1,2	3
1,2,3	3
1,2,3,4	3
2	6
2,3	6
…	

Closed contiguous sequential pattern set (min_sup = 2)

Closed contiguous sequential pattern	Support
1,2,3,4	3
2,3,4	6
2,3,4,5	3
6,2,3,4	2

Figure 1. Illustration of different type of sequential patterns. Each sequence represents the route traversed by a trajectory. Only part of the sequential pattern set and CSP set are shown due to space limit.

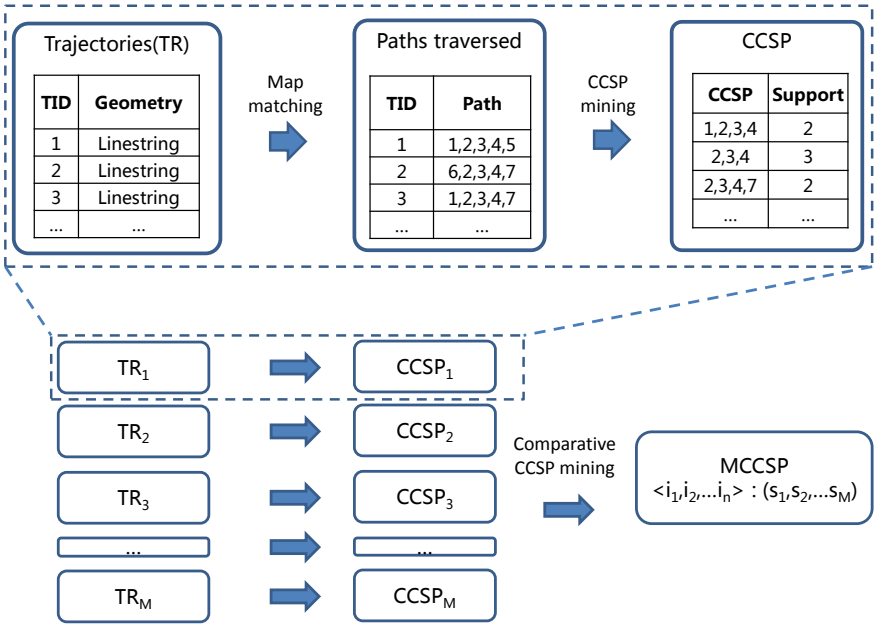


Figure 2. Illustration of comparative CCSP mining

over the input sets. With the input of temporally partitioned trajectories, the algorithm generates temporal route visit profiles. where clustering is performed to identify various temporal trends. Experiments on real-world taxi trajectory data demonstrate the efficiency and effectiveness of the approach.

2. Methodology

2.1. Comparative mining of CCSP sets

The comparative CCSP mining process is illustrated in Figure 2. Given M sets of trajectories $TR_1, \ldots TR_M$, map matching are performed to infer the path traversed by each trajectory and subsequently M CCSP sets can be mined with a minimum support of min_supp. The result are denoted by $CCSP_1, \ldots, CCSP_M$. The problem of *comparative mining* is formulated as finding the support s_m of *each* CSP p in $CCSP_m$ such that p is *frequent* ($s_m \geq min\_supp$) in *at least one* of the

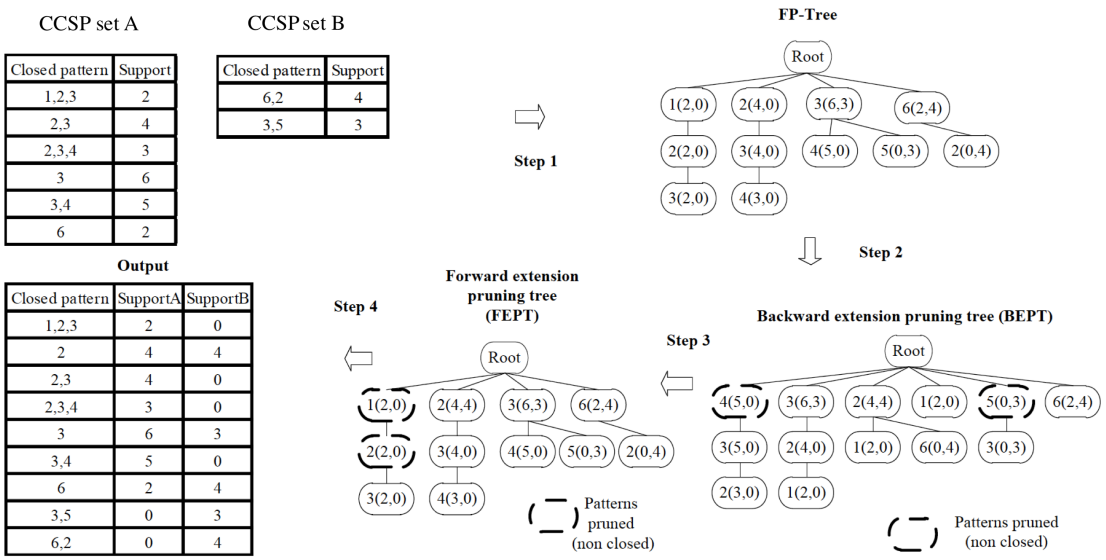


Figure 3. Illustration of comparative CCSP mining algorithm

input CCSP sets. The output is a multiple-support CCSP (MCCSP) set where each pattern is in form of

$$\langle i_1, i_2, \cdots, i_n \rangle (s_1, s_2, \cdots, s_M)$$

The multiple-support can be regarded as a *profile* of the CSP. An example of comparing two CCSP sets is illustrated in Figure 3. The result contains all CCSPs that occur in at least one of the input set with two support values.

Comparative mining can be much more complicated than expected due to the closedness constraint in CCSP set. The problem confronts two challenges:

1. some CCSPs only occur in one input set, which need to be maintained during the mining

2. some CCSPs are not explicitly contained but hidden in a CCSP due to the closedness constraint, which need to be regenerated

Taking the same example in Figure 3 for illustration, the CCSP $\langle 1, 2, 3 \rangle : 2$ only occurs in CCSP set A whereas the CCSP $\langle 2 \rangle$ is hidden in the two input sets because of the closedness constraint. The problem becomes more complicated when comparing more than two CCSP sets.

In this paper, an algorithm is designed to solve the comparative CSP mining problem by extending the bidirectional pruning based closed contiguous sequential pattern mining algorithm (BP-CCSM) developed by Yang and Gidófalvi (2018b). BP-CCSM employs three tree structures to create partitions of input sequences and CSPs to efficiently prune patterns by comparing nodes in a tree. In this paper, the proposed algorithm extends BP-CCSM in the following aspects:

1. Multiple supports are maintained at each node in the three trees to record the frequency of CSPs in each input set.

2. In the first tree, CSPs are inserted instead of sequences in BP-CCSM.

3. Pruning with the closedness constraint is performed by comparing multiple supports.

The process is illustrated with an example in Figure 3 where two CCSP sets are compared. Benefiting from the suffix partitioning of CSPs in the backward extension pruning tree and prefix partitioning of CSPs in the forward extension pruning tree, the algorithm efficiently solves the comparative mining problem. The algorithm can be slightly adjusted to store M supports at each node in case of M input CCSP sets.

2.2. Clustering of route visit profiles

When the M input trajectory sets are partitioned by time, e.g., hour of day, the output of comparative mining of CCSP sets can be interpreted as the temporal visit profile for all frequently traversed routes in the road network. In practice, the profiles can exhibit various trends together with a lot of noise. An example can be found in Figure 4 (a). To extract these trends, DBSCAN (Ester *et al.*, 1996) is selected to cluster these profiles. Compared with partitioning-based clustering approaches such as K-Means, DBSCAN does not require a predefined number of clusters and no constraint is imposed on the cluster shape. Given two route visit profiles with multiple-support of $P_a = (s_1^a, s_2^a, \cdots, s_M^a)$ and $P_b = (s_1^b, s_2^b, \cdots, s_M^b)$, the similarity measurement is defined as

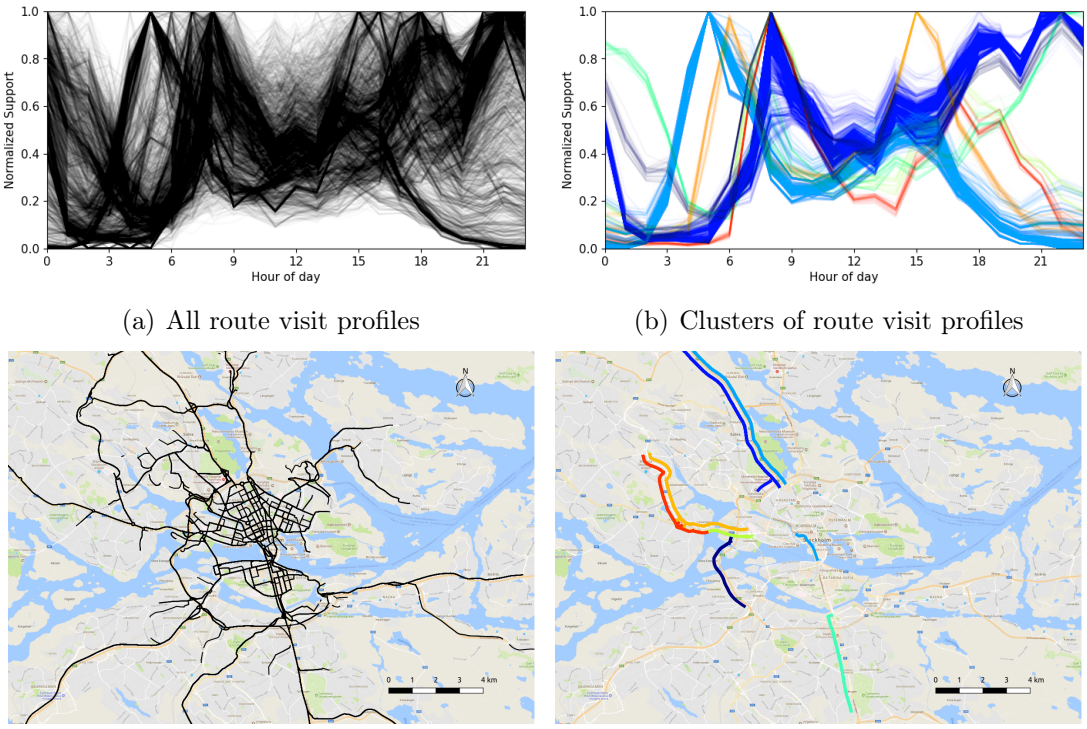
$$dist(P_1, P_2) = \sqrt{\sum_{1 \leq i \leq M} (\frac{1}{max(P_1)} s_i^a - \frac{1}{max(P_2)} s_i^b)^2}$$

which can be interpreted as the Euclidean distance between the normalized profile. The normalization is performed as many profiles show a similar shape with different magnitudes.

3. Preliminary results

The comparative mining algorithm is evaluated on the one-month taxi trip dataset used in Yang and Gidófalvi (2018a). It contains about 640,000 trips, which are partitioned into 24 subsets by hour of day (HOD). Each subset is matched to the road network using the Fast Map Matching algorithm proposed in Yang and Gidófalvi (2018a) then mined with BP-CCSM Yang and Gidófalvi (2018b). Consequently, 24 CCSP sets are generated and comparative mining is performed. The final result contains the hourly route visit profiles (24 supports) in the network, as displayed in Figure 4 (a) and (c). Several clusters can be observed from the profile distribution.

By empirically selecting DBSCAN configurations with distance threshold of 0.2 and minimum number of points of 80, seven clusters are discovered, as shown in Figure 4 (b) and (d). The clustering result is reasonable and can be explained by the network infrastructure of Stockholm. In Figure 4(d), there are two clusters distributed in the northern region of the city (dark and light blue color) that cover the highway connecting the Arlanda airport with the city center. Figure 4(b) shows that their profiles vary significantly because people catching their flight should leave earlier in the morning whereas the peak on the route from the airport to city center is on 9 am when most of the flights arriving at the airport. Another group



(a) All route visit profiles

(b) Clusters of route visit profiles

(c) Spatial distribution of all routes

(d) Spatial distribution of clusters

Figure 4. Hourly profile of routes. All routes are shown in (a) and (c) whereas only clustered routes are shown in (b) and (d).

of clusters, in brown and red colors, can be observed at the central area connecting the city center with another airport in Stockholm, which shows a different profile.

Another interesting observation is that routes with similar temporal profile are also spatially close to each other. It implies that the inflows and outflows of all the routes inside a cluster are temporally stable. Comparing Figure 4 (c) and (d), it can be observed that some regions have a lot of traffic but no cluster is formed. That phenomenon can be related with both the network infrastructure such as intersections and the underlying travel demand. Verification of the hypothesis is planned as a future work. Moreover, the large number of routes which are spatially close with very similar profile imply that the redundancy of the current result can be further reduced, which will also be investigated.

4. Conclusion and future work

This paper designed a comparative CCSP mining algorithm which efficiently and effectively identified temporal variations in route visiting frequencies from a large collection of trajectories. Future work is planned in the interpretation of clusters of route visit profiles, further reduction of redundancy and application of the algorithm in more accurate travel demand modeling and network infrastructure study.

References

Ester, M., *et al.*, 1996. A Density-based Algorithm for Discovering Clusters a Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *In*: *Proceedings of KDD'96*, 226–231.

Giannotti, F., *et al.*, 2007. Trajectory pattern mining. *In*: *Proceedings of KDD '07*, 330–339.

Li, X., *et al.*, 2012. Prediction of urban human mobility using large-scale taxi traces and its applications. *Frontiers of Computer Science in China*, 6 (1), 111–121.

Toole, J.L., *et al.*, 2015. The path most traveled: Travel demand estimation using big data resources. *Transportation Research Part C: Emerging Technologies*, 58, 162–177.

Yang, C. and Gidófalvi, G., 2018a. Fast map matching, an algorithm integrating hidden Markov model with precomputation. *International Journal of Geographical Information Science*, 32 (3), 547–570.

Yang, C. and Gidófalvi, G., 2018b. Mining and visual exploration of closed contiguous sequential patterns in trajectories. *International Journal of Geographical Information Science*, 32 (7), 1282–1304.

Ye, Y., *et al.*, 2009. Mining Individual Life Pattern Based on Location History. *In*: *International Conference on Mobile Data Management: Systems, Services and Middleware*, 1–10.

Zheng, Y., 2015. Trajectory Data Mining: An Overview. *ACM Transactions on Intelligent Systems and Technology*, 6 (3), 1–41.