Co-Clustering Network-Constrained Trajectory Data

Mohamed Khalil El Mahrsi, Romain Guigourès, Fabrice Rossi, Marc Boullé

Co-Clustering Network-Constrained Trajectory Data

Mohamed K. El Mahrsi, Romain Guigourès, Fabrice Rossi, and Marc Boullé

Abstract Recently, clustering moving object trajectories kept gaining interest from both the data mining and machine learning communities. This problem, however, was studied mainly and extensively in the setting where moving objects can move freely on the euclidean space. In this paper, we study the problem of clustering trajectories of vehicles whose movement is restricted by the underlying road network. We model relations between these trajectories and road segments as a bipartite graph and we try to cluster its vertices. We demonstrate our approaches on synthetic data and show how it could be useful in inferring knowledge about the flow dynamics and the behavior of the drivers using the road network.

1 Introduction

Monitoring traffic on road networks is generally handled using dedicated sensors that provide estimations of the number of vehicles traversing the road portion on which they are deployed. Due to their prohibitive installation and maintenance costs, the deployment of these sensors is mainly limited to the primary road network (i.e. highways and main arteries). Consequently, the road network's state reported using

Mohamed K. El Mahrsi
Télécom ParisTech - Département Informatique et Réseaux, 46 Rue Barrault 75634 Paris CEDEX 13 France, e-mail: khalil.mahrsi@telecom-paristech.fr

Romain Guigourès
Orange Labs, 2 Avenue Pierre Marzin 22300 Lannion France, e-mail: romain.guigoures@orange.com

Fabrice Rossi
Équipe SAMM EA 4543, Université Paris I Panthéon-Sorbonne, 90 Rue de Tolbiac 75634 Paris CEDEX 13 France, e-mail: fabrice.rossi@univ-paris1.fr

Marc Boullé
Orange Labs, 2 Avenue Pierre Marzin 22300 Lannion France, e-mail: marc.boulle@orange.com

this kind of solutions is partial and incomplete which complicates the application of data mining tasks that aim to extract meaningful knowledge about flow dynamics and mobility patterns.

Thanks to the advances in the fields of telecommunication and geo-positioning, an alternative approach may consist in taking advantage of GPS logs collected on moving objects that are equipped with ad hoc devices (such as smartphones). These logs can be acquired through dedicated data acquisition campaigns (using probing vehicles, buses, taxis, etc.), through crowdsourcing mechanisms in which users contribute their own trajectories, etc. Trajectory data can thus be harvested on a large scale which helps provide a better coverage of the road network compared to sensor data.

Clustering is a widely used technique in exploratory data analysis. Given a set of observations, cluster analysis consists in partitioning these observations into groups (called clusters) in such fashion that objects belonging to the same group are more similar to each other (w.r.t. a given criterion) than to objects from other groups. Most prior work on trajectory clustering focused on the case of moving objects evolving freely in the euclidean space [Kalnis et al., 2005, Benkert et al., 2006, Lee et al., 2007, Jeung et al., 2008]. Often in real applications, however, moving objects must comply with the existence of an underlying network (for instance, vehicles evolve in the road network, airplanes must remain in invisible but well defined air corridors, etc.). The topological constraints imposed by this network play a key role in determining the similarity between trajectories and should logically be accounted for during the clustering process. Clustering network-constrained trajectories gained in popularity only recently with the publication of work such as [Kharrat et al., 2008, Liu et al., 2008, Roh and Hwang, 2010], etc.

The insightful idea of using a graph-based approach to cluster trajectory data was first used in [Guo et al., 2010]. We built upon the premisses of this idea in [El Mahrsi and Rossi, 2012b] where we used a graph representation to model the similarity relationships between trajectories and clustered this similarity graph to extract clusters of trajectories that exhibit similar mobility patterns. This approach was extended in [El Mahrsi and Rossi, 2012a] and used to regroup similar road segments that can eventually be used to further enhance the interpretability of trajectory clusters. In the present work, we retain this idea of using a graph representation as we model the interactions between trajectories and road segments using a bipartite graph and we study two different approaches to clustering its vertices.

The remainder of this paper is organized as follows. Our data model and proposed approaches are presented in Sect. 2. Section 3 illustrates our experimental study where we demonstrate our propositions' capacity to highlight and discover interesting trajectory and road segment clusters. Related work is briefly discussed in Sect. 4. Finally, conclusions and future work are presented in Sect. 5.

2 Clustering Approaches

In the network-constrained case, trajectories are often modeled using a symbolic data model [Kharrat et al., 2008, Lou et al., 2009, Roh and Hwang, 2010]. Each trajectory T is represented as a series of succeeeding road segments (this is done by applying a map-matching algorithm, such as [Lou et al., 2009], to the original GPS logs). Therefore, two entities are eligible for applying clustering techniques: (i) trajectories, and (ii) road segments.

We model the data as a bipartite graph $\mathscr{G} = (\mathscr{T}, \mathscr{S}, \mathscr{E})$. $\mathscr{T} = \{T_1, T_2, ..., T_n\}$ is the dataset of trajectories, $\mathscr{S} = \{s_1, s_2, ..., s_m\}$ is the set of all the road segments composing the road network that registered at least one traversal, and $\mathscr{E}$ is the set of edges modeling interactions between trajectories and road segments (i.e. an edge e exists between a trajectory T and a road segment s if and only if T visited s at least once). This representation is illustrated in Fig.1 depicting five trajectories T_1, T_2, T_3, T_4, and T_5 interacting with eight road segments and the corresponding bipartite graph.

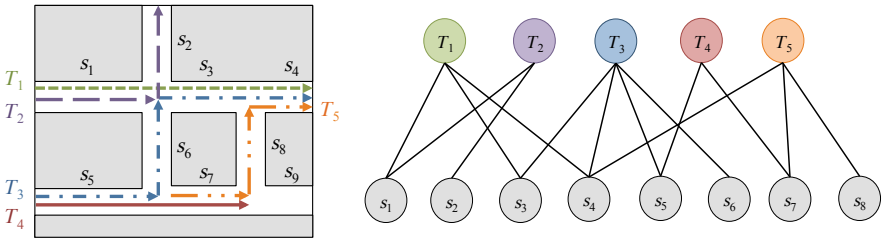


Fig. 1 A bipartite graph is used to model interactions between the trajectories and the road network's segments. Each trajectory and each road segment is represented as a vertex in the graph. Edges are created between each trajectory and the set of road segments it visited.

We will first attempt to project the bipartite graph $\mathscr{G}$ on both its trajectory vertices $\mathscr{T}$ and road segment vertices $\mathscr{S}$ and study clustering the resulting graphs separately (Sect. 2.1). Secondly, we will process $\mathscr{G}$ directly using a co-clustering approach (Sect. 2.2).

2.1 Clustering the Projected Trajectory and Segment Graphs

Our bipartite graph $\mathscr{G}$ is composed of two types of vertices, trajectory vertices and road segment vertices. We can project $\mathscr{G}$ on the set of trajectory vertices $\mathscr{T}$ which produces a new, simple graph $\mathscr{G}_{\mathscr{T}} = (\mathscr{T}, \mathscr{E}_{\mathscr{T}}, \mathscr{W}_{\mathscr{T}})$ that represents similarity relationships between trajectories. In this setting, $\mathscr{T}$ stands for vertices representing trajectories, $\mathscr{E}_{\mathscr{T}}$ are edges indicating the presence of similarities between pairs of trajectories (an edge $e_{\langle T_i, T_j \rangle}$ exists between two trajectories T_i and T_j if they share

at least one common road segment), and $\mathscr{W}_{\mathscr{T}}$ are weights assigned to edges based on the strength of the similarity between trajectories they connect. An example of a projected trajectory similarity graph is depicted in Fig. 2.

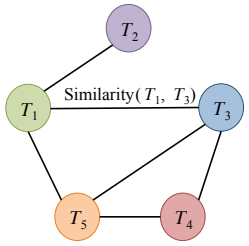


Fig. 2 The trajectory similarity graph resulting from the projection of the bipartite graph depicted in Fig. 1 on its trajectory vertices. Here, each trajectory is represented as a vertex and weighted edges inter-connect trajectories based on their similarity.

The most basic weighting strategy is to assign to each edge $e_{\langle T_i,T_j\rangle}$ a weight $\omega_{\langle T_i,T_j\rangle}$ that is equal to the count of common road segments between the two trajectories T_i and T_j. However, the main drawback of this approach is that it completely neglects the spatial properties of road segments (for instance, a very short road segment and a lengthy one have equal contributions to the similarity). Therefore, we proposed a more sophisticated and spatially-aware weighting strategy in [El Mahrsi and Rossi, 2012b]. It is this strategy that we will continue to use here.

For each road segment s visited by a trajectory T, we calculate its contribution (w.r.t. this trajectory) based not only on its spatial length but also on the frequency of its appearance in the dataset. This contribution is basically an adaptation of tf-idf (term frequency - inverse document frequency) weighting widely used in information retrieval, modified to account for spatiality. The contribution $w_{s,T}$ of segment s to trajectory T is calculated according to Eq. (1).

$$w_{s,T} = \frac{n_{s,T} \cdot \text{length}(s)}{\sum_{s' \in T} n_{s',T} \cdot \text{length}(s')} \cdot \log \frac{|\mathscr{T}|}{|\{T_i : s \in T_i\}|} \,. \tag{1}$$

$n_{s,T}$ is the number of times the trajectory T visited the road segment s (usually equal to 1), length(s) is the spatial length of the segment. $|\mathscr{T}|$, the total number of trajectories in $\mathscr{T}$, and $|\{T_i : s \in T_i\}|$, the number of trajectories that visited s. The second term in $w_{s,T}$ is used to penalize frequently-traveled road segments (with the intuition that the more a segment is traveled, the less it is relevant w.r.t. similarity evaluation and vice versa).

We evaluate the similarity between pairs of trajectories using a cosine similarity and we assign the weights in $\mathscr{G}_{\mathscr{T}}$ accordingly (2):

$$\omega_{\langle T_i,T_j\rangle} = \frac{\sum_{s \in \mathscr{S}} w_{s,T_i} \cdot w_{s,T_j}}{\sqrt{\sum_{s \in \mathscr{S}} w_{s,T_i}^2} \cdot \sqrt{\sum_{s \in \mathscr{S}} w_{s,T_j}^2}} \,. \tag{2}$$

By analogy, we can project the bipartite graph $\mathscr{G}$ on the set of road segment vertices $\mathscr{S}$ in which case we obtain a segment similarity graph $\mathscr{G}_{\mathscr{S}} = (\mathscr{S}, \mathscr{E}_{\mathscr{S}}, \mathscr{W}_{\mathscr{S}})$. In this graph, a similarity edge $e_{\langle s_i, s_j \rangle}$ indicates that at least one trajectory visited both road segments s_i and s_j. Here again, it is totally feasible to assign edge weights based solely on the count of common trajectories, but instead we define a weighting technique based on trajectory relevance [El Mahrsi and Rossi, 2012a] similarly to what we did earlier when processing trajectories. The similarity between two road segments s_i and s_j is expressed as follows (3):

$$\omega_{\langle s_i, s_j \rangle} = \frac{\sum_{T \in \mathscr{T}} w_{T,s_i} \cdot w_{T,s_j}}{\sqrt{\sum_{T \in \mathscr{T}} w_{T,s_i}^2} \cdot \sqrt{\sum_{T \in \mathscr{T}} w_{T,s_j}^2}} \ . \tag{3}$$

With :

$$w_{T,s} = \frac{n_{s,T}}{\sum_{T' \in \mathscr{T}} n_{s,T'}} \cdot \log \frac{|\mathscr{S}|}{|s' \in \mathscr{S} : s' \in T|} \ . \tag{4}$$

The first part of $w_{T,s}$ evaluates the "contribution" (or importance) of trajectory T to the road segment s by calculating the ratio between the number of visits $n_{s,T}$ made by T to s and the number of visits s received from all the trajectories in $\mathscr{T}$. The second parts evaluates the overall relevance of T based on comparing the number of different segments it visited $|s' \in \mathscr{S} : s' \in T|$ to the total number of road segments $|\mathscr{S}|$.

We propose to cluster the projected trajectory and segment graphs separately in order to discover trajectory clusters on one side and road segment clusters on the other. To do so, we chose to apply modularity-based community detection using an algorithm that implements the directives described in [Noack and Rotta, 2009]. This choice is mainly motivated by the fact that vertices in such similarity graphs are expected to have high degrees and modularity-based clustering is reputed to outshine other approaches in such settings. Nevertheless, we do not exclude the use of other graph clustering algorithms (e.g. spectral clustering [Meila and Shi, 2000]) if these can yield better results.

The used algorithm produces a hierarchy of nested (trajectory or segment) clusters that are suitable for multi-level exploration where the user can start by inspecting a few, coarse clusters in order to quickly understand the general motion trends then proceed to exploring clusters of interest with higher levels of detail by means of successive refinements. Also, once the trajectory and segment partitions are found, they can either be analyzed separately or cross-compared and interpreted based on each other.

Given a dataset of n trajectories that travelled on a road network composed of m segments, the time complexity for clustering the trajectory graph is theoretically in $O(n^3)$ whereas clustering road segments is done in $O(m^3)$ [Noack and Rotta, 2009].

2.2 Direct Co-Clustering of the Bipartite Graph

We now propose to study clustering the bipartite graph $\mathscr{G}$ directly. To achieve this end, we apply a co-clustering approach to the graph's adjacency matrix. In the adjacency matrix, trajectories are represented in the rows whilst road segments are represented on the columns. The intersection of row i with column j indicates the number of times trajectory T_i visited the road segment s_j ($1 \leq i \leq n$ and $1 \leq j \leq m$). Co-clustering works by rearranging the rows and columns of the adjacency matrix in order to highlight blocs that have homogeneous density. These blocs are then used to derive two partitions simultaneously (one partition for trajectories and the other for road segments in our case). A co-clustering structure, that we refer to as $\mathscr{M}$ hereafter, is usually defined through a set of modeling parameters. Ours are described in Table 1.

Table 1 Notations.

Bipartite graph $\mathscr{G}$	**Co-clustering model $\mathscr{M}$**
$\mathscr{T}$: set of trajectories	$C_{\mathscr{T}}$: set of trajectory clusters
$\mathscr{S}$: set of road segments	$C_{\mathscr{S}}$: set of road segment clusters
$\mathscr{E} = \mathscr{T} \cap \mathscr{S}$: set of traversals of road segments in $\mathscr{S}$ by trajectories in $\mathscr{T}$	$C_{\mathscr{E}} = C_{\mathscr{T}} \cap C_{\mathscr{S}}$: co-clusters of trajectory and road segments

The objective of co-clustering algorithms is to infer the best partition of the bipartite graph. By applying such approaches, trajectories are regrouped if they travel along common road segments and, vice versa, road segments are clustered together if they are visited by the same trajectories. The main advantage of these techniques is that they do not require preprocessing nor do they require the definition of an "artificial" similarity between trajectories or between segments. Nonetheless, they do present the drawback of being computationally expensive.

We opt for the MODL [Boullé, 2011] approach to conduct the co-clustering of $\mathscr{G}$. We made this choice because this approach (i) is non-parametric and does not require user intervention or fine-tuning, (ii) is easily scalable and can, consequently, be used to analyze large datasets, and (iii) was already and successfully applied to geo-tagged data [Guigourès et al., 2012]. In MODL, a quality criterion is defined according to a Maximum A Posteriori (MAP) approach (5):

$$\mathscr{M}^* = \underset{\mathscr{M}}{\operatorname{argmax}} P(\mathscr{M}) P(\mathscr{D}|\mathscr{M}) \ . \tag{5}$$

First, an a priori probability $P(\mathscr{M})$ is defined based on the data (denoted $\mathscr{D}$). This probability tries to characterize each of the modeling parameters of the model $\mathscr{M}$ by assigning to each one of them a penalty (which corresponds to their minimal coding length, calculated based on descriptive statistics of the data). Next, the likelihood of the data given the data model $P(\mathscr{D}|\mathscr{M})$ is defined. The likelihood measures the cost of re-encoding $\mathscr{D}$ with the parameters of $\mathscr{M}$. Consequently, the most likely co-clustering model is the one that is most faithful to the original data (in other terms, the likelihood tends to favor relevant and informative structures). Retrieving the best

co-clustering (i.e. the one optimizing the global criterion $\mathscr{M}^*$) consists in realizing the best trade-off between conciseness and accuracy.

Since co-clustering problems are often *NP*-complete, the clustering is conducted using an agglomerative greedy heuristic. Initially, the trivial, most refined model is considered (this model contains only one trajectory and one road segment per cluster). Then, all cluster merging operations are evaluated and the best merge is applied (if it results in a decrease of the quality criterion). Once no more merging operations are possible, the result of the heuristic is refined using a post-optimization step in which some elements swap their cluster memberships. The whole process is encapsulated within a VNS (Variable Neighborhood Search, [Hansen and Mladenovic, 2001]) metaheuristic that restarts the algorithm several times with different random cluster initializations. Full details and a thorough evaluation of the MODL approach can be retrieved in [Boullé, 2011].

MODL has a complexity of $O(|\mathscr{E}|\sqrt{|\mathscr{E}|}\log(|\mathscr{E}|))$ where $\mathscr{E}$ indicates the total number of edges in the bipartite graph $\mathscr{G}$ (which, in our case, translates to the overall number of road segment traversals). This complexity, however, is only observed in the worst and very unlikely case where each trajectory in the dataset $\mathscr{T}$ visits each single road segment in the road network.

3 Experimental Study

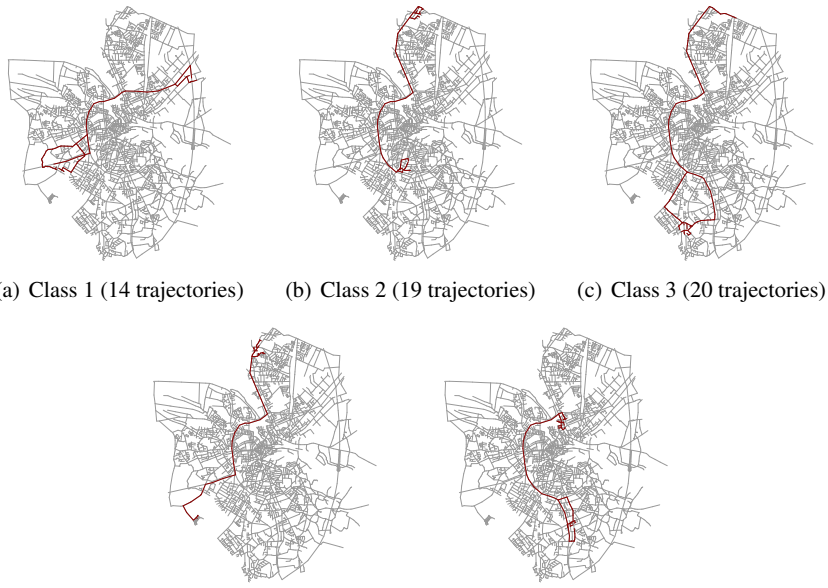
In this section, we demonstrate how the proposed approaches can be used to discover and analyze motion patterns in road networks. Our experimental setting is described in Sect. 3.1 whereas results and their interpretation are presented in Sect. 3.2 and Sect. 3.3.

3.1 Experimental Setting

In order to test our propositions, we use synthetic datasets of labeled trajectories. These datasets are generated intentionally to contain trajectories that are supposed to form natural clusters using the following strategy. The space covered by the road network (represented by the minimum bounding rectangle regrouping all of its vertices) is divided into a grid of equally-sized rectangular cells (or zones). For each of the clusters to be generated, a zone is selected randomly and all of its contained vertices are chosen to play the role of departure points. Similarly, a second zone (different from the first) is also selected randomly and its vertices are used as arrival points. For each trajectory to be included in the cluster, a departure (resp. arrival) vertex is drawn randomly from the set of departure (resp. arrival) vertices and the trajectory is generated as the set of road segments forming the shortest path linking the two vertices (the shortest path calculation is based on travel time and takes into account the characteristics of the visited road segments such as speed limitations, etc.).

The number of trajectories in each cluster is fixed randomly in-between two minimum and maximum user-defined values. The data generation process is conceived in such fashion that interactions between clusters can occur (examples of interactions include clusters converging from different departure zones to a common arrival zone, inverted clusters where the departure zone of one cluster is the arrival zone of the other and vice versa, etc.).

Since this experimental study is intended to showcase how our approaches can contribute to discovering meaningful knowledge about mobility in the road network, we make do with a case study involving a small dataset composed of 85 trajectories. These trajectories are spread across five distinct clusters (depicted in Fig. 3) and visited 485 road segments in total. We designate these original clusters as "classes" hereafter in order to distinguish them from those that will be retrieved using the clustering algorithms. The dataset is generated using the Oldenburg road network's graph (originally provided with the Brinkhoff generator[Brinkhoff, 2002]) which is composed of 6105 vertices (i.e. road intersections) and 14070 directed edges (i.e. road segments).



(a) Class 1 (14 trajectories) (b) Class 2 (19 trajectories) (c) Class 3 (20 trajectories)

(d) Class 4 (20 trajectories) (e) Class 5 (12 trajectories)

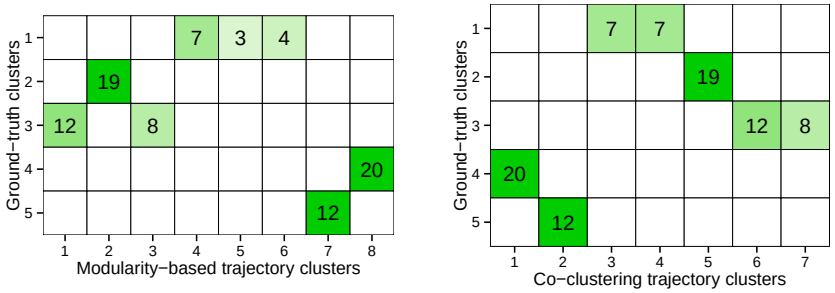
Fig. 3 Original classes in the dataset. Some of the classes present natural interactions. For instance, classes 2 and 3 start from the same departure zone and travel together for a given portion then diverge to different arrival zones. They also interact with class 1 in the central portion of the road network.

3.2 Analysis of Trajectory Clusters

Applying modularity-based clustering to the projected trajectory similarity graph produces a partition containing three clusters (in contrast with the original five classes). This is mainly due to the fact that some classes interact considerably. Their interactions were not visible when labeling the classes individually during the data generation process. The clustering algorithm, however, was able to detect these interactions and regroup the trajectories accordingly.

Since the algorithm we apply is a hierarchical algorithm that produces a multi-level hierarchy of nested clusters, we can further refine the clusters in a given level by exploring their subsequent clusters. In the case of the dataset at hand, the second level reveals the existence of eight trajectory clusters. The confusion matrix between these clusters and the original classes is illustrated in Fig. 4(a). In this level, all the clusters contained in the partition are pure. Three of the original five classes were retrieved flawlessly whereas the two remaining clusters were further refined (class 1 was divided into three clusters and class 3 into two). This "over-partitioning" is legitimate and can be justified considering the variability of the trajectories contained in the concerned classes.

Co-clustering, on the other hand, directly retrieves a partition that is faithful to the original data (cf. Fig. 4(b)). Here again two original classes were over-partitioned and three classes were retrieved correctly. Since the results (w.r.t. trajectory clusters) of applying MODL directly to the bipartite graph resemble those obtained by modularity-based clustering of the projected trajectory graph, it is more logical to use the former since it contains no preprocessing contrary to the latter where similarity calculations need to be performed first.

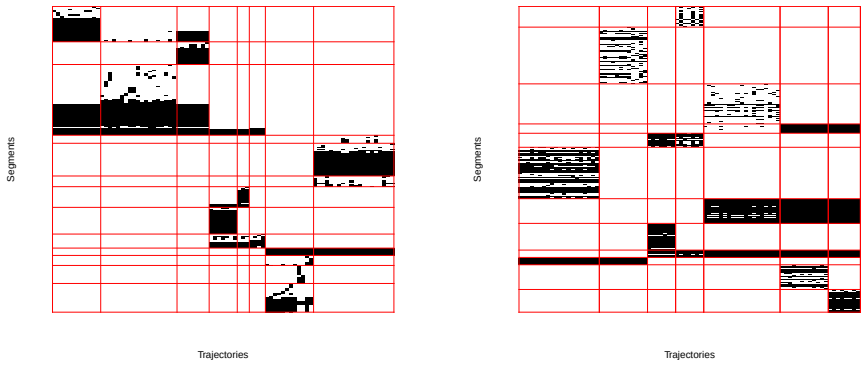


(a) Modularity (b) Co-Clustering

Fig. 4 Confusion matrices of the original classes (ground-truth clusters) in the data and the clusters discovered by applying (a) the modularity-based approach and (b) the MODL co-clustering approach (cells are color-coded based on the ratio of the ground-truth cluster they contain). The modularity-based approach yields an Adjusted Rand Index (ARI) of 0.849, whereas MODL yields a slightly higher (i.e. better) ARI value of 0.862.

3.3 Mutual Analysis of Trajectory and Segment Clusters

Let's now study the adjacency matrix of the original bipartite graph $\mathscr{G}$. We re-ordered the rows and columns of the matrix in order to bring together trajectories and segments belonging to the same clusters (cf. Fig. 5). We observe in the case of the projected graphs (Fig. 5(a)) that road segments are regrouped together based on common trajectories without accounting for the traffic's volume. Therefore, road segments that are rarely visited can be attached to segments that are visited frequently. This translates, when looking at the adjacency matrix, into the presence of blocs with heterogeneous distributions in which some segments are travelled by all the trajectories in the cluster whereas others are only visited by a limited subset of trajectories. In co-clustering, on the other hand, segments are correlated based on usage which results in blocs of homogeneous densities (Fig. 5(b)).



(a) Modularity (b) Co-clustering

Fig. 5 Crossed matrices of trajectory clusters (columns) and road segment clusters (rows) retrieved through (a) modularity-based clustering and (b) co-clustering.

By inspecting trajectory clusters and road segment clusters simultaneously, it is possible to characterize road segments based on the roles they play in traffic.This makes it possible to identify hubs that are frequently travelled by multiple groups of vehicles transiting to different regions (Figure 6(a)), secondary roads (Figure 6(b)), and even rarely frequented alleys. Therefore, our methodology makes it possible to characterize the topological structure of the underlying road network based on trajectories contributing their usage information.

Mutual information is frequently-used in co-clustering to quantify the correlations between partitions of the studied variables. These are, in our case, trajectories and road segments. Mutual information is always positive. High values of this metric usually indicate that trajectory clusters visit rather exclusive and unique segment clusters. We use mutual information in our study to quantify the relationship between

(a) A highway hub

(b) Secondary roads leading to peripheral areas of the city

Fig. 6 Example of road segment clusters.

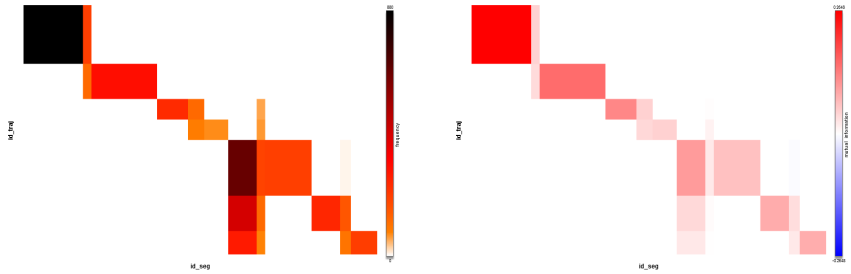
pairs of trajectory and segment clusters and their contribution to the model's mutual information. Given a cluster of trajectories c_T and a cluster of road segments c_S, the contribution of the pair to mutual information, denoted $mi(c_S, c_T)$, is calculated as follows (6).

$$mi(c_S, c_T) = P(c_S, c_T) \log \frac{P(c_S, c_T)}{P(c_S)P(c_T)} \ . \tag{6}$$

Where $P(c_S, c_T)$ is the probability of a segment traversal to belong to a trajectory in c_T and covering a road segment that belongs to c_S, $P(c_S)$ is the probability of visiting a segment belonging to c_S, and $P(c_T)$ is the probability of having a trajectory belonging to c_T.

A positive contribution to mutual information indicates that the number of visits of trajectories in c_T to road segments in c_S is higher than what is expected in case the two clusters were completely independent one from the other. Vice versa, a negative contribution is an indicator that quantity of traffic w.r.t. is inferior to normal. Finally, a null contribution to mutual information indicate that traffic either conforms to what is expected or is very low.

Figure 7(b) presents the contribution to mutual information for each couple of co-clusters discovered in the dataset at hand. For instance, if we take the left, top-most co-cluster we can notice that the segments are exclusively travelled by members of a single trajectory cluster and that, vice versa, trajectories in this trajectory cluster travel almost uniquely on the members of this segment cluster. In this case, the trajectory cluster comprises 21.6% of the studied trajectories and the road segments cluster 17.3% of segments in the dataset. If we suppose that both clusters are independent, then we can expect no more than observing $21,6\% \times 17.3\% = 3.7\%$ of the total road segment traversals to be originating from both clusters. Here, however, we observe that no less than 17.3% from the total traversals belong to this co-cluster which largely exceeds the expected traffic in case of unrelated and independent clusters.



(a) Frequency (b) Mutual information

Fig. 7 Frequency and mutual information of the retrieved co-clusters.

Notice that the mutual information contributes an information that is different from the frequency matrix. We can observe that some road segment clusters are significantly traversed by members belonging to multiple trajectory clusters. This behavior is quite characteristic of hubs that vehicles coming from different regions cross in order to attend different destinations. Some of these clusters have very small contrast w.r.t. mutual information which indicates that traffic on the hub is rather balanced.

4 Related Work

Approaches to trajectory clustering are mainly adaptations of existing algorithms to the case of trajectories. These include moving clusters [Li et al., 2004], flock patterns [Benkert et al., 2006], convoy patterns [Jeung et al., 2008], the TRACLUS partition-and-group framework [Lee et al., 2007], etc. The aforementioned algorithms use euclidean-based similarities and distances and disregard the presence of an underlying network. Therefore, they can be used only in the case of unconstrained trajectories. The insightful idea of using a graph-based approach to cluster trajectory data was first introduced in [Guo et al., 2010]. The approach is applied to free moving trajectories and considers the latter as sets of GPS points. Unlike our graph-based approaches, the authors do not rely on an underlying network as the basis of similarity calculations.

The first attempt to study the similarity between network-constrained trajectories is reported in [Hwang et al., 2005]. The proposition requiers a priori knowledge of points of interest in the road network and cannot, consequently, be used in an unsupervised learning context. An extension of moving clusters to network-constrained trajectories is presented in [Liu et al., 2008]. Roh et Hwang [Roh and Hwang, 2010] present a network-aware approach to clustering trajectories where the distance between trajectories in the road network is measured using shortest path calculations. A baseline algorithm, using agglomerative hierarchical clustering, as well as a more efficient algorithm, called NNCluster, are presented for the purpose of regrouping the

network constrained trajectories. In [Kharrat et al., 2008], the authors describe an approach to discovering "dense paths" or sequences of frequently traveled segments in a road network. The approach is extended in [Kharrat et al., 2009] to study the temporal evolution of dense paths.

Our approaches differ from existing propositions on two key aspects. First, the majority of existing work use density-based algorithms that require fine-tuning of their parameter values and assume that trajectories in the same cluster have a rather homogeneous density (which is rarely the case as discussed in [Roh and Hwang, 2010]). In contrast, we opt for non-parametric algorithms that rely on robust and well defined clustering quality criteria. Secondly, existing approaches often use flat clustering, thus producing a unique level of clusters that can be overwhelming to analyse in the case of large datasets. Our propositions produce hierarchies of nested clusters that are suitable for multi-level exploration: the user can start with a small number of clusters to quickly understand the macro-organization of flow dynamics in the road network, then proceed to refining clusters of interest to reveal more details.

5 Conclusions

In this paper, we studied clustering network-constrained trajectory data from the angle of a bipartite graph clustering problem. We notably studied this problem from two different perspectives. At first, we considered the problem as a community detection problem conducted separately on two simple graphs (one depicting resemblances between trajectories and the other depicting similarities between road segments). Then we proceeded to co-cluster the bipartite graph directly to automatically retrieve clusters of interacting trajectories and road segments.

The main contribution of this work resides in its methodology of applying graph-based techniques to trajectory data in order to extract clusters describing mobility patterns in road networks. These clusters can be used by experts and road planners in conjunction with other data sources in order to understand trafic and driver behaviors. The applied clustering algorithms (modularity-based clustering used on the projected graphs and MODL used for co-clustering the bipartite graph) were essentially used to showcase and illustrate the interestingness of our problem formulation. As such, they can be replaced by other graph clustering and co-clustering approaches.

In future work, we will mainly focus on experimenting with the co-clustering technique on a larger scale (bigger datasets) as well as in the presence of noisy data.

References

[Benkert et al., 2006] Benkert, M., Gudmundsson, J., Hübner, F., and Wolle, T. (2006). Reporting flock patterns. In *ESA'06: Proceedings of the 14th conference on Annual European Symposium*, pages 660–671, London, UK. Springer-Verlag.

[Boullé, 2011] Boullé, M. (2011). Data grid models for preparation and modeling in supervised learning. In *Hands-On Pattern Recognition: Challenges in Machine Learning, vol. 1*, pages 99–130. Microtome.

[Brinkhoff, 2002] Brinkhoff, T. (2002). A framework for generating network-based moving objects. *Geoinformatica*, 6:153–180.

[El Mahrsi and Rossi, 2012a] El Mahrsi, M. K. and Rossi, F. (2012a). Graph-Based Approaches to Clustering Network-Constrained Trajectory Data. In *Proceedings of the Workshop on New Frontiers in Mining Complex Patterns (NFMCP 2012)*, pages 184–195, Bristol, Royaume-Uni.

[El Mahrsi and Rossi, 2012b] El Mahrsi, M. K. and Rossi, F. (2012b). Modularity-Based Clustering for Network-Constrained Trajectories. In *Proceedings of the 20-th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2012)*, pages 471–476, Bruges, Belgique.

[Guigourès et al., 2012] Guigourès, R., Boullé, M., and Rossi, F. (2012). A triclustering approach for time evolving graphs. In *ICDM Workshops*.

[Guo et al., 2010] Guo, D., Liu, S., and Jin, H. (2010). A graph-based approach to vehicle trajectory analysis. *J. Locat. Based Serv.*, 4:183–199.

[Hansen and Mladenovic, 2001] Hansen, P. and Mladenovic, N. (2001). Variable neighborhood search: Principles and applications. *European Journal of Operational Research*, 130(3):449–467.

[Hwang et al., 2005] Hwang, J.-R., Kang, H.-Y., and Li, K.-J. (2005). Spatio-temporal similarity analysis between trajectories on road networks. In *ER (Workshops)*, Lecture Notes in Computer Science, pages 280–289. Springer.

[Jeung et al., 2008] Jeung, H., Shen, H. T., and Zhou, X. (2008). Convoy queries in spatio-temporal databases. In *ICDE '08: Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, pages 1457–1459, Washington, DC, USA. IEEE Computer Society.

[Kalnis et al., 2005] Kalnis, P., Kalnis, P., Mamoulis, N., and Bakiras, S. (2005). On discovering moving clusters in spatio-temporal data. *IN SSTD*, pages 364–381.

[Kharrat et al., 2008] Kharrat, A., Popa, I. S., Zeitouni, K., and Faiz, S. (2008). Clustering algorithm for network constraint trajectories. In *SDH*, Lecture Notes in Geoinformation and Cartography, pages 631–647. Springer.

[Kharrat et al., 2009] Kharrat, A., Popa, I. S., Zeitouni, K., and Faiz, S. (2009). Caractérisation de la densité de trafic et de son évolution à partir de trajectoires d'objets mobiles. In Menga, D. and Sedes, F., editors, *UbiMob*, volume 394 of *ACM International Conference Proceeding Series*, pages 33–40. ACM.

[Lee et al., 2007] Lee, J.-G., Han, J., and Whang, K.-Y. (2007). Trajectory clustering: a partition-and-group framework. In *SIGMOD '07: Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 593–604, New York, NY, USA. ACM.

[Li et al., 2004] Li, Y., Han, J., and Yang, J. (2004). Clustering moving objects. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 617–622, New York, NY, USA. ACM.

[Liu et al., 2008] Liu, W., Wang, Z., and Feng, J. (2008). Continuous clustering of moving objects in spatial networks. In *KES '08: Proceedings of the 12th international conference on Knowledge-Based Intelligent Information and Engineering Systems, Part II*, pages 543–550, Berlin, Heidelberg. Springer-Verlag.

[Lou et al., 2009] Lou, Y., Zhang, C., Zheng, Y., Xie, X., Wang, W., and Huang, Y. (2009). Map-matching for low-sampling-rate gps trajectories. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '09, pages 352–361, New York, NY, USA. ACM.

[Meila and Shi, 2000] Meila, M. and Shi, J. (2000). Learning Segmentation by Random Walks. In *NIPS*, pages 873–879.

[Noack and Rotta, 2009] Noack, A. and Rotta, R. (2009). Multi-level algorithms for modularity clustering. In *Proceedings of the 8th International Symposium on Experimental Algorithms*, SEA '09, pages 257–268, Berlin, Heidelberg. Springer-Verlag.

[Roh and Hwang, 2010] Roh, G.-P. and Hwang, S.-w. (2010). Nncluster: An efficient clustering algorithm for road network trajectories. In *Database Systems for Advanced Applications*, volume 5982 of *Lecture Notes in Computer Science*, pages 47–61. Springer Berlin - Heidelberg.