Profiling users of the Vélo'v bike sharing system (for extended abstract submission)

Albrecht Zimmermann AZIMMERMAN@INSA-LYON.FR
INSA-Lyon, CNRS, LIRIS UMR5205, F-69621, France

Mehdi Kaytoue MEHDI.KAYTOUE@TAPASTREET.COM
Tapastreet Ltd.,36 Dame Street, Dublin, Ireland

Marc Plantevit MPLANTEV@LIRIS.CNRS.FR
Université Lyon 1, CNRS, LIRIS UMR5205, F-69622, France

Céline Robardet CROBARDE@LIRIS.CNRS.FR
INSA-Lyon, CNRS, LIRIS UMR5205, F-69621, France

Jean-François Boulicaut JFBOULIC@LIRIS.CNRS.FR
INSA-Lyon, CNRS, LIRIS UMR5205, F-69621, France

Abstract

Detecting and characterizing geographical areas that are attractive places for specific people, in specific contexts, is an important but challenging new problem. Mobility traces and their related circumstances can be modeled thanks to an augmented graph in which nodes denote geographic locations and edges are represented by a set of transactions that describe users' demographic information (e.g. age, gender, etc.) as well as the conditions of the movement (e.g. day/night, holiday, transportation mode, etc.). We propose to extract connected subgraphs that are related to some user profiles, and use it to understand the usages of the Vélo'v bike sharing system.

1. Introduction

With the rapid development of wireless sensor technologies in mobile environments, such as GPS, Wi-Fi and RFID, it is now very easy to monitor people mobility and use this information to provide personalized services. The last decade has witnessed a huge growth in the analysis of mobility (Giannotti & Pedreschi, 2008). These studies focus only on mining trajectories and their applications (Li et al., 2010; Luo et al., 2013; Monreale et al., 2009; Wang et al., 2011; Zheng et al., 2009), but do not take into account the contextual information of the individual trajectories. Inherent contexts of the trajectories are essential data to produce accurate and valuable models of mobility patterns. Contexts and trajectories encode complementary knowledge that cannot be deduced from one another.

The problem considered hereafter is how to detect and characterize geographical areas that are attractive places and routes for specific contexts. Such areas are frequently accessed together in certain conditions by users of similar profiles compared to all contexts and users. Starting from a relational database that gathers information on people movements – such as origin, destination, date and time of travel, means of transport, reasons for traveling, etc. – as well as demographic data, we adopt a graph-based representation that results from the aggregation of individual travels. In such a graph, the vertices are locations or points of interest (POI) and the edges stand for user's co-visitations. Travel information as well as user demographics are labels associated to the edges of the graph.

Figure 1 (a) depicts an example of travels undertaken by users (denoted $u_1, \ldots, u_4$). For each user, we know her age and gender, the context of the move (here the time period during which the travel takes place – day or night) and the set of movements, identified by a pair origin/destination, that occur in this context. Capital letters, from A to E, represent POI. This table can also be viewed as an edge-attributed graph where edges stand for movements and are labeled by the attribute values of the context. For instance, we have a directed edge (A, B) labeled by $(F, 20, Day)$ for

Proceedings of the 2^{nd} International Workshop on Mining Urban Data, Lille, France, 2015.

User	Gender	Age	Time	Travels
u_1	F	20	Day	(A,B), (A,C), (C,B)
u_1	F	20	Night	(D,C),(D,E),(E,A), (E,D)
u_2	M	23	Day	(A,B),(B,C),(C,A), (C,B)
u_2	M	23	Night	(A,B),(B,C),(C,B) (C,D),(D,C),(D,E), (E,D)
u_3	F	45	Day	(A,B),(B,C),(C,D), (D,A),(D,E),(E,D)
u_3	F	45	Night	(B,D),(D,B)
u_4	M	50	Day	(A,B),(B,C),(C,B), (C,D),(D,A),(D,E), (E,D)
u_4	M	50	Night	(A,C),(C,A)

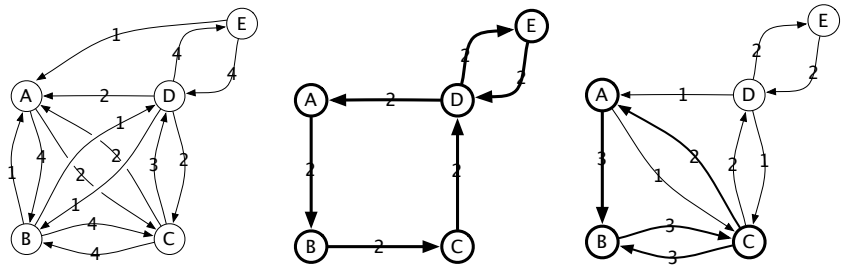


Figure 1. Example of contextualized trajectories: (a) Transactional view; (b) Aggregate graph w.r.t the most general context $\star = (Age \in [20, 50],\ Gender \in \{F, M\},\ Time \in \{Day,\ Night\})$; (c) Aggregate graph w.r.t. context $(Age \in [45, 50], Time = Day)$; (d) Aggregate graph w.r.t. context $(Gender = M)$;

the user u_1. Given a specific context, the edge-attributed graph can be transformed into an aggregate graph the edges of which are weighted by the number of attributed edges that hold for the context. Three examples of aggregated graphs are given in Figure 1 (b),(c) and (d). The weights of the aggregated graph can be seen as the support of the context in the graph.

The problem is thus to identify the contexts and sub-graphs that are specific to one another. By specific, we mean that a large proportion of the weight of each sub-graph edge mainly corresponds to users that satisfy the context. The adequacy of a context to an edge is assessed by a χ^2 test and some novel quality measures that makes it possible to identify the so-called *demographic and contextualized specific areas (DCSA)*. Two *DCSA* patterns are presented in Figure 1 (c) and (d) (in bold): The first one identifies a sub-graph that is traveled during the *day* , mainly by people with *age greater than* 45. In the second sub-graph, bold edges are very specific to *male* persons' behavior, whatever the travel time.

Such an approach provides new insights to understand urban data. To illustrate this fact, we consider its use on operating data of the bike sharing system of Lyon called Vélo'v.

2. Mining Contexts and trajectories

To analyze the interdependence between people movements and demographics, we model the data as an edge-attributed graph that is aggregated with respect to a context. This operation results in a graph the weights of which are the amount of users that, in the same time, follow the corresponding edges and satisfy the context. Comparisons between pairs of aggregate graphs makes it possible to identify demographic and contextualized specific sub-graphs that are specific to a context. These different notions are defined below.

2.1. Edge-attributed graphs and their aggregates

Edge-Attributed Graphs are able to model various information networks. In the specific ones we consider hereafter, a set of transactions, defined over a finite set of attributes, is associated to each edge:

Definition 1 (Edge-Attributed Graph) *An edge-specific attributed graph G is a graph denoted $G = (V, E, A, T)$ where V is a set of vertices, $E \subseteq V \times V$ is a set of edges, $A = \{A_1, \ldots, A_m\}$ is a set of m edge-specific attributes and $T \subseteq E \times dom(A_1) \times \ldots \times dom(A_m)$ is a set of edge-specific transactions, i.e., each transaction t of T is a tuple $(e, a_1, \ldots, a_m)$ with $e \in E$, $a_i \in dom(A_i)$, that depicts a contextual edge e. T_e denotes the subset of edge-specific transactions involving the edge e.*

A context $C = (A'_1, \ldots, A'_m)$, with $A'_i \subseteq dom(A_i)$, is a domain restriction over the attributes associated to the edges. As the attributes can be of different types (e.g., symbolic, numerical, ordinal), the subset A'_i are convex to avoid uninterpretable results. The transactions associated to e that satisfy the context C are denoted $T_e(C)$ with $T_e(C) = \{t \in T_e | \forall a_i \in t, a_i \in A'_i\}$.

Inspired by the hyper graph cube model defined in (Wang et al., 2014), the transactions of T can be aggregated as in traditional data cubes. Transactions that satisfy a context C are grouped by edges, and an aggregation function is used as weight of the edges of the graph. In the following definition of aggregate graph, we use the *count* aggregation function.

Definition 2 (Aggregate graph) *Given a context $C = (A'_1, \ldots, A'_m)$, and an attributed graph $G = (V, E, A, T)$, the aggregation of G by C results in a weighted graph $G_C = (V, E_C, W_C)$ where*

- *E_C is the set of edges e such that there exists at least*

one transaction in T_e that holds for the context C, i.e., $E_C = \{e \in E | T_e(C) \neq \emptyset\}$

- *$W_C(e)$ is a weight associated to each edge of E_C that is equal to the number of transactions associated to e that are generalized by C, i.e. $W_C(e) = |T_e(C)|$.*

It is generally necessary to focus the analysis only on edges the weight of which is large enough. To this end, $G_C^{\theta_W}$ is the thresholded graph whose edges have a weight greater than or equal to a threshold θ_W:

$$G_C^{\theta_W} = (V, E_C^{\theta_W}, W_C), with E_C^{\theta_W} = \{e \in E_C \mid W_C(e) \geq \theta_W\}$$

Figure 1(b) illustrates the aggregate graph with respect to the context $C = (dom(Sex), dom(Age), dom(Time))$, denoted $\star$ to be consistent with database notations, while Figures 1(b) and (c) show the aggregate graphs $G_{(\{F,M\},[45,50],\{Day\})}$ and $G_{(\{M\},[20,50],\{Day,Night\})}$.

2.2. Context specific edges

To assess the specificity of a context C to an edge e, we consider the proportion of users of the edge e that satisfy the context and propose to statistically assess this value by a Pearson's chi-squared test of independence.This test determines whether or not the context appears more specifically in the edge e than in the whole graph.

A user might satisfy or not a context C, and follow or not the edge e. These four possible outcomes are denoted **C** and $\overline{\mathbf{C}}$, **e** and $\bar{\mathbf{e}}$. Table 1 is the contingency table O that collects the observed outcomes of **e** and **C**. The null hypothesis states that the occurrences of the outcomes **e** and **C** are statistically independent. If we suppose that **C** occurs uniformly over all the edges of the graph, there are $W_\star(e)\frac{\sum_{x\in E} W_C(x)}{\sum_{x\in E} W_\star(x)}$ chances that a user that satisfies the context **C** context follows the edge e. The three others outcomes under the null hypothesis are constructed on the same principle and are given in the contingency table E presented in table 2. The value of the statistical test is thus

$$X^2 = \sum_{i=1}^{2}\sum_{j=1}^{2} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

The null distribution of the statistic is approximated by the χ^2 distribution with 1 degree of freedom, and for a significance level of 5%, the critical value is equal to $\chi^2_{0.05} = 3.84$. Consequently, X^2 has to be greater than 3.84 to establish that the weight related to a context on a given edge deviates sufficiently to reject the null hypothesis and conclude that the edge weight is biased at 95% significance level.

A point has to be stressed here: the rejection of the null hypothesis can be due to either a very large or a very low value of $|T_e(C)|$. The distinction between these two cases will be done thanks to an additional measure q defined in the subsequent section. To be considered significant for a context C, an edge must have a positive value on q. θ_q is thus a positive threshold used to estimate the specificity of C on the each edge. The resulting significant aggregate graph is denoted $G_C^{\chi^2}$ in the following.

2.3. Pattern definition and general mining task

Demographic and contextualized specific areas (*DCSA*) are connected components of $G_C^{\chi^2}$. This set of patterns is denoted $CC_C^{\chi^2}$. Not all such patterns are of interest for an end-user and we propose to take into account the end-user's particular interest by considering additional constraints. However, doing so often leads to well-known thresholding issues that limit the effective application of such an approach. Indeed, most existing constraints are defined by a parameter whose value is highly dependent on the application and/or the dataset. It is therefore hard for an end-user to decide on an adequate value. To overcome this problem, we do not consider such constraints but instead take the user's preferences into account. Therefore, we propose to discover patterns that maximize the end-user preferences, with the possibility to set additional thresholds.

We denote by M a set of convex measures used by the end-user as preferences over the patterns in $CC_C^{\chi^2}$. Each measure gives a real value to each pattern and the mining task consists to retrieve all the *DCSA* patterns that maximize those measures collectively. Therefore, given a set of preferences $M = \{m_1, \ldots, m_k\}$, the patterns that are part of the Pareto front – also called skyline – i.e., that are not dominated by another one.

Definition 3 (Dominance) *Given a set of preferences $M = \{m_1, \ldots, m_k\}$, P_d* dominates *P_s, denoted $P_d >_M P_s$, iff $\forall i : m_i(P_d)) \geq m_i(P_s)$ and $\exists j : m_j(P_d) > m_j(P_s)$.*

The mining task that we address is the following:

Problem 1 (Discovery of *DCSA* patterns.) *Given an edge-specific attributed graph $G = (V, E, A, T)$, a set of user-preferences M, the problem of demographic and contextualized specific areas (DCSA) pattern mining is to compute the* skyline *of DCSA patterns defined by:*

$$\{(C, (X, Y)) \mid (X, Y) \in CC_C^{\chi^2} \text{ and } \nexists (C', (X', Y')) \in CC_{C'}^{\chi^2} \text{ such that } (C', (X', Y')) >_M (C, (X, Y))\}$$

The main challenge arising from this problem setting is that we have to consider all possible contexts as well as the connected components of the significant aggregate graph $G_C^{\chi^2}$.

	$\mathbf{e}$	$\overline{\mathbf{e}}$	
$\mathbf{C}$	$W_C(e)$	$\sum_{x\in E} W_C(x) - W_C(e)$	$\sum_{x\in E} W_C(x)$
$\overline{\mathbf{C}}$	$W_\star(e) - W_C(e)$	$\sum_{x\in E} W_\star(x) - W_\star(e) - \sum_{x\in E} W_C(x) + W_C(e)$	$\sum_x W_\star(x) - \sum_x W_C(x)$
	$W_\star(e)$	$\sum_{x\in E} W_\star(x) - W_\star(e)$	$\sum_{x\in E} W_\star(x)$

Table 1. Contingency table O of events $\mathbf{C}$ and $\mathbf{e}$.

	$\mathbf{e}$	$\overline{\mathbf{e}}$	
$\mathbf{C}$	$W_\star(e)\frac{\sum_{x\in E} W_C(x)}{\sum_{x\in E} W_\star(x)}$	$\left(\sum_{x\in E} W_\star(x) - W_\star(e)\right) \times \frac{\sum_{x\in E} W_C(x)}{\sum_{x\in E} W_\star(x)}$	$\sum_{x\in E} W_C(x)$
$\overline{\mathbf{C}}$	$W_\star(e) \times \left(1 - \frac{\sum_x W_C(x)}{\sum_{x\in E} W_\star(x)}\right)$	$\left(\sum_{x\in E} W_\star(x) - W_\star(e)\right) \times \left(1 - \frac{\sum_{x\in E} W_C(x)}{\sum_{x\in E} W_\star(x)}\right)$	$\sum_{x\in E} W_\star(x) - \sum_{x\in E} W_C(x)$
	$W_\star(e)$	$\sum_{x\in E} W_\star(x) - W_\star(e)$	$\sum_{x\in E} W_\star(x)$

Table 2. Contingency table E under the null hypothesis.

The search space is thus at the same time very large and difficult to traverse.

Notice that we have given a generic formalization of the *DCSA* pattern discovery problem. In the following, we will instantiate the set of user preferences M with several measures to be maximized. In particular, we will introduce the quality measure q used to identify highly specific edges of a context and discuss it.

3. *DCSA* Discovery

3.1. Quality measures

For a given context C, we aim to identify the edges whose number of users satisfying the context C is greater than what would be expected based on the information encoded in the whole graph. Generally speaking, a measure that can be used to assess such behavior would subtract the relative weight of the edge e – the weight of the edge normalized by some term – in the whole aggregate graph from its relative weight in the context aggregate graph: $\frac{W_C(e)}{W_C^{norm}} - \frac{W_\star(e)}{W_\star^{norm}}$. Obviously, if this term is larger than 0 then the edge's weight is greater than expected considering the other edges in the graph. This means that this edge is of relatively greater importance for the context than it is for the full graph.

The question that remains is which terms to use for normalization. A first option would consist of adopting the same solution as in the χ^2-test, that is using the sum over the weights of all the edges for the context: $\frac{W_C(e)}{\sum_{x\in E} W_C(x)} - \frac{W_\star(e)}{\sum_{x\in E} W_\star(x)}$. We will refer to these normalization terms as $W^{\Sigma}_{[C|\star]}$ from here on. A semantic interpretation of this choice is to consider the edge weights as supports and identifying the edges that have unexpectedly high support on C.

A second option instead normalizes by the maximal weight of any edge matching the context: $\frac{W_C(e)}{\max_{x\in E} W_C(x)} - \frac{W_\star(e)}{\max_{x\in E} W_\star(x)}$, referred hereafter as $W^{max}_{[C|\star]}$. In this case, the semantic is more similar to the one of multi-label classification or exceptional model mining: the context describes a set of instances, and each edge corresponds to a *label* of that context. The goal is to identify labels for which this context is more descriptive than expected.

In both cases, it is easy to achieve very high scores by simply drive down W_c^{norm} until $\frac{W_C(e)}{W_C^{norm}}$ comes close to 1.0 for all involved edges. To avoid this kind of over-fitting, we introduce a normalizing factor, which penalizes contexts that are too specific:

$$q(e, C) = \frac{W_C^{norm}}{W_\star^{norm}} \times \left(\frac{W_C(e)}{W_C^{norm}} - \frac{W_\star(e)}{W_\star^{norm}}\right)$$

Therefore, we consider that an edge is over-expressed in a context C iff $q(e, C) > 0$ and under-expressed iff $q(e, c) < 0$. As in our problem setting the goal is to find edges that are specific to a context, we enforce that every edge in a *DCSA* pattern $(C, (X, Y))$ is over-expressed for the context C. The same measure and approach could, however, be used to identify edges that are *avoided* by certain populations.

This quality measure is applied to every edge of a *DCSA* pattern $P = (C, (X, Y))$. To compute the score of the whole pattern P, we have to aggregate the individual scores. This aggregation can be done in different ways. When calculating the score of P, we can sum over all edges from Y: $q^{\Sigma}(C, (X, Y)) = \sum_{e\in Y} q(e, C)$, compute the arithmetic mean of the edge scores: $\overline{q} = \frac{q^{\Sigma}(C,(X,Y))}{|Y|}$, or the standard deviation $q^{\sigma} = \sqrt{\frac{\sum_{e\in Y}(\overline{q} - q(e,C))^2}{|Y|-1}}$.

3.2. User preferences

To identify interesting patterns without requiring the end-user to set thresholds, we are considering the patterns that belong to the Pareto front defined by the set of user preferences M. Naturally, the quality measure belongs to M. Furthermore, the end-user may be interested in maximizing q^{Σ} or $\overline{q}$. The user's preferences can involve some measures on the whole pattern $(C,(X,Y))$ as the quality measures but also some measures that focus on a specific part of the pattern, especially the graph structure (X,Y). Therefore, the end-user may be interested in maximizing the number of vertices $m_v = |X|$ or the number of edges $m_e = |Y|$. In the rest of the paper, we take into account these preferences to compute the skyline of *DCSA* patterns (i.e., $M = \{q^{\Sigma}, \overline{q}, m_v, m_e\}$). We define our algorithm with this set of preferences but any other convex measures can be taken into account as a user preferences in our approach. This point is discussed later.

3.3. Optional threshold constraints

Essentially, our approach only requires as input a set of user preferences to look for *DCSA* patterns in the edge-attributed graph G. Nevertheless, the end-user may also want to constrain the shape of the *DCSA* patterns. To this end, it is possible to specify some additional constraints in our approach by defining some minimum thresholds w.r.t. some measures.

For instance, the end-user may add some conditions on the weight on the edges for privacy or interpretability reasons. To this end, she can specify a minimum threshold θ_w. Note, that this threshold is not related to a measure that explicitly appears in the set of user preferences M. It is also possible in our approach to add some minimum threshold constraints on the measures that are taken into account as preference in M, especially to discard some extrema *DCSA* patterns in the skyline, i.e., to avoid reporting patterns that are in the skyline because they have an extreme value regarding one preference measure, while under-performing for the other ones. The use of these optional thresholds makes it possible to obtain a much narrower skyline of *DCSA* patterns. A minimum threshold on a measure m_i from M is denoted θ_{m_i}.

4. Algorithm

The theoretical search space of *DCSA* patterns is structured as a lattice which contains all possible combinations of contexts and edge sets. However, the collection of edges E_C (and the subgraph induced by these edges) is determined by the context C (in interplay with optional thresholds). It is noteworthy that the set of vertices V_C involved in a pattern $P = (C,(V_C,E_C))$ can be directly derived from E_C. Therefore, we do not mention V_C in the following. This means, for instance, that the lattice is bounded on one end by $\{\star, E_\star\}$ instead of $\{\star, V \times V\}$ and that there is no single bound at the other end but instead a number of *DCSA* patterns involving maximally specific contexts C, i.e. the most stringent domain restrictions and their respective edge sets E_C.

The enumeration of all the patterns by materializing and traversing all possible bi-sets from the lattice is not feasible in practice. Therefore, the algorithm enumerates contexts in a depth-first search manner. Then the optional constraints and the computation of upper bounds on the quality measure(s) in the set of user preferences are used to reduce the search space while using their properties to not develop unpromising candidates. The enumeration can be represented as a tree where each node is an enumeration step. A node consists of a pattern bi-set P identifying the pattern (C, E_C) and a set of candidates $Cand$. P is the pattern in construction. $P.C$ denotes the domain restrictions on the edge-attributes, i.e., the context, and the context of any pattern generated from P specializes $P.C$. $P.E$ denotes all edges involved in $G_{P.C}$. $Cand$ contains the possible extension of P, i.e., the set of restrictions that can be added to P. At the beginning, $P = \{\star, E_\star\}$ and $Cand$ is the set of all possible domain restrictions. Each node of the enumeration tree has up to $|Cand|$ children, depending on pruning effects.

Let us now describe the upper bounds used to drastically reduce the search space. Even if the quality measure q is not anti-monotonic, we can compute an upper bound of it when using W^{max}. Specifically, for a given edge e with weight $W_{P.C}(e)$, the maximal possible edge score, the upper bound, is $ub(e, P.C) = \frac{W_{P.C}(e)}{W_\star^{max}} \times \left(1 - \frac{W_\star(e)}{W_\star^{max}}\right)$. This result is obtained in a similar way to what was proposed in the literature for convex measures.

The upper bound used for q^{Σ} evaluated on a connected component is obtained by adding up the upper bounds of its individual edges. Those individual upper bounds can also be ordered in descending order to derive upper bounds for $\overline{q}$. Since the largest upper bound for $\overline{q}$ will typically correspond to relatively low q^{Σ}, m_v, and m_e, a series of upper bounds $\{u_{q^\Sigma}, u_{\overline{q}}, u_{m_v}, u_{m_e}\}$ are calculated that trade off against $u_{\overline{q}}$ against the others.

Finally, if $\overline{q} \notin M$, we can compute a tighter upper bound on q^{Σ} by sorting all edges' $W_C(e)$ in descending order and dynamically updating two sums of edge scores – one for weights below the a split point, one for weights above – while observing that W_C^{max} needs to be consistent for all edges; something that is not enforced for individual upper bounds.

For q using W^{Σ}, to the best of our understanding no upper

bound can be computed and therefore no pruning is performed.

We can overload the *dominance* notion and say that a *DCSA*$(C,(X,Y))$ dominates a pattern P if for all its sets of upper bounds $\{ub_{q^\Sigma}, ub_{\overline{q}}, ub_e, ub_v\}$:

$$(C,(X,Y)) >_M P \Leftrightarrow$$
$$(q^\Sigma(X,Y) \geq ub_{q^\Sigma} \wedge \overline{q}(X,Y) \geq ub_{\overline{q}} \wedge |X| \geq ub_{m_v} \wedge$$
$$|Y| \geq ub_{m_e} \wedge (q^\Sigma(X,Y) > ub_{q^\Sigma} \vee \overline{q}(X,Y) > ub_{\overline{q}}$$
$$\vee |X| > ub_{m_v} \vee |Y| > ub_{m_e}))$$

Any P the upper bounds of which are dominated by the current result set can be safely pruned. Since all singleton contexts are enumerated in the first step, we have information we can use for pruning $Cand$ as well.

5. Travel patterns in the VÉLO'V system

VÉLO'V is the bicycle sharing and renting system run by the city of Lyon (France) and the company JCDecaux.[1] There is in total 348 VÉLO'V stations across the city of Lyon. The VÉLO'V dataset contains movement data collected in a 2 year period (Jan. 2011 – Dec. 2012). Each movement includes both bicycle stations and timestamps for departure and arrival, as well as some basic demographics about the user of the bike. We considered only movements made by registered users, and aggregated all movements a user performed between any two stations for the entire time period. Hence, the VÉLO'V stations are nodes in the graph (342 in total), and edges link two stations if a VÉLO'V customer checked out a bicycle at the first station and returned it at the second one. We treat the edges as undirected. Customers are described by nominal attributes such as gender, type of membership card, ZIP code and country of residence, as well as a numerical one: year of birth. There are a total $50,601$ customers. The data set comprises around $2,000,000$ contextualized edges in total.

5.1. Problem setting and parameter choices

The problem setting for our experiments on the VÉLO'V data is essentially the one that we outlined in the introduction to motivate our work: given the characteristics of different users, we aim to identify populations that use the rental bicycles in a particular manner. Given our earlier results on the synthetic data, we limit ourselves to q with a normalization factor, both with the sum and maximum as normalizing weights. We use $M = \{q^\Sigma, \overline{q}, u_e, u_v\}$ and fix $\theta_{u_e} = 9$, $\theta_{u_v} = 10$, $\theta_W = 10$ to find interesting enough patterns. We also performed runs using q^σ, to gain further insight into the effect of using this user preference. The clearest effect of adding q^σ can be seen in the number of patterns found (Table 3).

[1] http://www.velov.grandlyon.com/

W^{max}		W^Σ	
$q^\sigma \in M$	$q^\sigma \notin M$	$q^\sigma \in M$	$q^\sigma \notin M$
1760 (9)	56 (6)	5124	116

Table 3. Number of *DCSA*with and without q^σ and after post-processing (in parentheses)

Number of patterns and redundancy reduction. In both cases, the effect of the skyline operator leads to redundancy: a pattern that has slightly lower q^Σ, for example, than another one yet involves one more edge will neither dominate nor be dominated. To tackle this problem, we implemented a simple post-processing operation inspired by those used in local pattern mining, or rule learning (Liu et al., 1998):

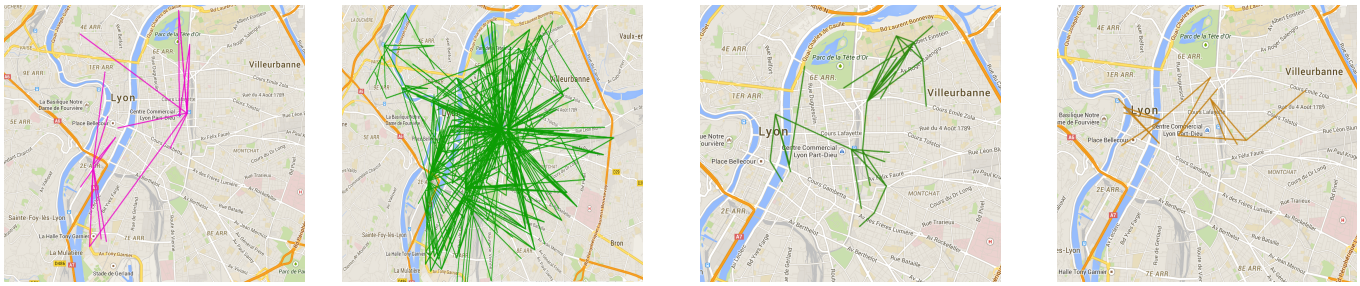
1. All patterns are sorted according to a quality criterion (we chose $\overline{q}$ to begin with those patterns the edges of which are particularly strongly expressed).
2. Each pattern is considered in turn and if it adds at least $\theta_{pp} \in (0,1)$ additional nodes or edges proportional to the current set of patterns, it is selected.

A first realization that this process brought is that using W^Σ might not work for a data set as large as VÉLO'V: individual edge scores are so relatively low that $\overline{q}$ becomes very low and sorting virtually impossible. For W^{max}, however, the post-processing scheme can be applied and we find that using q^σ mainly increases the number of redundant patterns.

Qualitative results. Table 3 also lists the number of patterns remaining for W^{max} ($\theta_{PP} = 0.5$) and we present some of those patterns in the following section. Notably, this simple scheme allows us to select patterns that collectively cover the majority of nodes or edges, respectively. Figure 2 shows 4 different *DCSA* from VÉLO'V. Pattern (i) identifies people born after 1968, living in a city (Saint Chamond) located approximately 50km from Lyon. It is therefore not surprising that the edges involve the two main train stations of Lyon: Perrache (south-west) and Part-Dieu (center), from which users take bicycles to areas that are not easily reached by metro or tram, such as the 1st and 4th arrondissements.

The edges of pattern (ii) radiate from all of Lyon's train stations, not only the major ones. Its description refers to holders of a regional train subscription (monthly or yearly), and the pattern notably involves 200 nodes, almost sixty percent of the total. It is therefore very likely that this is a pattern that identifies commuters.

Pattern (iii) is somewhat harder to interpret. It involves users born in or after 1980 and we can identify three main



(i) YoB $\geq 1968, \text{ZIP} = 42400$ **(ii)** YoB $\geq 1962, \text{CAT} = \text{OURA}$ **(iii)** YoB $\geq 1980, \text{TYP} = \text{standard}$ **(iv)** YoB $\geq 1992, \text{ZIP} = 69003$

Figure 2. DCSA discovered from VÉLO'V

areas: the scientific campus in the north, the Presqu'ile and its pubs, and the shopping area in the center of Lyon. It is notable that several of the long edges correspond to very comfortable cycling routes: 1) the edge running south-north on the Presqu'ile probably corresponds to Rue de la République, a pedestrian zone that especially at night makes for nice cycling, and 2) the eastern bank of the Rhône offers nicely developed cycling lanes along the water, for instance.

Pattern (iv), finally, does not seem to be very exciting: young people that live in the 3^{rd} arrondissement use VÉLO'V bicycles to move around in their area. At a second glance, however, this is the closest that we will come to a ground truth in real-world data: the ZIP code of users aligns with the area where the bicycles are used!

6. Related Work

Finding descriptions of subpopulations for which the distribution of a pre-defined target value is significantly different from the distribution in the whole data is a problem that has been widely studied in *subgroup discovery* (SGD)(Lavrac et al., 2004). When the target is a set of attributes, the high dimensionality of the search space requires heuristic approaches (*exceptional model mining*, EMM) (Leman et al., 2008). In our case, the targets consist of relative weights of edges in connected components and are arguably dynamic: edges can become over- or under- expressed and connected components change accordingly.

Subspace graph clustering (Günnemann et al., 2010) considers attributed-node graphs and seeks to identify quasi-cliques whose nodes are similar in their attribute values while being densely connected. The algorithm proceeds by grouping nodes together according to shared attribute values and assessing whether they form a quasi-clique. Similarly to our work, the end result consists of subgraphs and a set of attributes on which those subgraphs exist. But while attribute values group nodes and edges remain unchanged in the former approach, attribute values determine the weights of edges with nodes unchanged in our setting.

In similar settings *trend mining in dynamic graphs* methods seeks to identify sets of nodes whose attribute values develop consistently over time (Desmier et al., 2013).

Several techniques aim at extracting *subgraphs in edge-attributed graphs* such as (Qi et al., 2012; Boden et al., 2012; Bonchi et al., 2012; Berlingerio et al., 2013). There are two conceptual differences between those approaches and our work: 1) they all use edge information (attributes or weights) to find *similar* edges, and 2) use those edges to group (and often partition) *nodes*. Contrary to this, we are interested in edges the relative weight of which *differs* from the full graph and consider those *edges* (and their description) their own reward. In (Qi et al., 2012), edges are considered *similar* according to similar collections of labels, and are used to partition nodes into communities, edge weights are not considered. Similarly, (Bonchi et al., 2012) tries and find clusters of edges (and therefore nodes) in which all edges have the *same* labels. From some of the same authors comes the proposal of *multidimensional* network analysis (Berlingerio et al., 2013). The authors formulate the idea that connections between nodes exist in different *dimensions*, e.g. cities can have both train and plane connections, and extend a number of network measures to multi-dimensional graphs. There is a semantic difference to our approach: two nodes having edges in a number of different dimensions are, for instance, considered to be more strongly connected. In our framework, a single edge between two nodes is enough – given that it has much higher relative weight for a given context than would have been expected.

The *multi-layer coherent subgraph* approach (Boden et al., 2012) uses numerical labels on nodes to assess edges' *similarity*. Those values depend on the layers of the graph. Nodes among which edges have similar edge-weights are assembled into quasi-cliques, additional layers added in which those nodes form quasi-cliques as well. There is again the conceptual contrast between *similar* weights and *difference* from background weights: MiMag might consider edges similar that are not very typical for a context/layer. There is also the semantic difference that all

edges in a *DCSA* match the describing context and we find those edges that are typical, whereas Boden *et al.* might group very different contexts and discover that individuals belonging to those contexts behave very similar for a certain set of nodes.

7. Conclusion

In this paper, we defined the problem of finding *DCSA* in edge-attributed graphs. This problem finds many applications, especially in location based social networks and recommendation systems: it allows to find connected components highly characteristic of a given category of users. We showed how an inductive approach rooted in machine learning (with an original set of quality measures in subgroup discovery) and database theory (with the skyline operator) can answer this challenging problem. This is achieved thanks to an efficient data-mining algorithm ESCARGot that avoids materializing all contexts/induced-graph pairs and benefits from pruning and upper bound computations techniques. We considered a case-study on urban data: the analysis of the bicycle sharing system Vélo'v of Lyon. In that context, we show that *DCSA* make it possible to provide new valuable insights.

References

Berlingerio, Michele, Coscia, Michele, Giannotti, Fosca, Monreale, Anna, and Pedreschi, Dino. Multidimensional networks: foundations of structural analysis. *WWW*, 16 (5-6):567–593, 2013.

Boden, Brigitte, Günnemann, Stephan, Hoffmann, Holger, and Seidl, Thomas. Mining coherent subgraphs in multi-layer graphs with edge labels. In *KDD*, pp. 1258–1266, 2012.

Bonchi, Francesco, Gionis, Aristides, Gullo, Francesco, and Ukkonen, Antti. Chromatic correlation clustering. In *KDD*, pp. 1321–1329, 2012. URL http://doi.acm.org/10.1145/2339530.2339735.

Desmier, Elise, Plantevit, Marc, Robardet, Céline, and Boulicaut, Jean-François. Trend mining in dynamic attributed graphs. In *ECML/PKDD*, pp. 654–669, 2013.

Giannotti, Fosca and Pedreschi, Dino. *Mobility, data mining and privacy*. Springer Science & Business Media, 2008.

Günnemann, Stephan, Färber, Ines, Boden, Brigitte, and Seidl, Thomas. Subspace clustering meets dense subgraph mining. In *ICDM*, pp. 845–850, 2010.

Lavrac, Nada, Kavsek, Branko, Flach, Peter A., and Todorovski, Ljupco. Subgroup discovery with CN2-SD. *Journal of Machine Learning Research*, 5:153–188, 2004.

Leman, Dennis, Feelders, Ad, and Knobbe, Arno J. Exceptional model mining. In *ECML/PKDD*, pp. 1–16, 2008.

Li, Zhenhui, Ji, Ming, Lee, Jae-Gil, Tang, Lu-An, Yu, Yintao, Han, Jiawei, and Kays, Roland. Movemine: Mining moving object databases. In *SIGMOD*, pp. 1203–1206. ACM, 2010. ISBN 978-1-4503-0032-2. doi: 10.1145/1807167.1807319. URL http://doi.acm.org/10.1145/1807167.1807319.

Liu, Bing, Hsu, Wynne, and Ma, Yiming. Integrating classification and association rule mining. In *KDD*, pp. 80–86, 1998.

Luo, Wuman, Tan, Haoyu, Chen, Lei, and Ni, Lionel M. Finding time period-based most frequent path in big trajectory data. In *SIGMOD*, pp. 713–724. ACM, 2013. ISBN 978-1-4503-2037-5. doi: 10.1145/2463676.2465287. URL http://doi.acm.org/10.1145/2463676.2465287.

Monreale, Anna, Pinelli, Fabio, Trasarti, Roberto, and Giannotti, Fosca. Wherenext: A location predictor on trajectory pattern mining. In *KDD*, pp. 637–646. ACM, 2009. ISBN 978-1-60558-495-9. doi: 10.1145/1557019.1557091. URL http://doi.acm.org/10.1145/1557019.1557091.

Qi, Guo-Jun, Aggarwal, Charu C., Tian, Qi, Ji, Heng, and Huang, Thomas S. Exploring context and content links in social media: A latent space method. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(5):850–862, 2012. doi: 10.1109/TPAMI.2011.191. URL http://doi.ieeecomputersociety.org/10.1109/TPAMI.2011.191.

Wang, Dashun, Pedreschi, Dino, Song, Chaoming, Giannotti, Fosca, and Barabasi, Albert-Laszlo. Human mobility, social ties, and link prediction. In *KDD*, pp. 1100–1108. ACM, 2011. ISBN 978-1-4503-0813-7. doi: 10.1145/2020408.2020581. URL http://doi.acm.org/10.1145/2020408.2020581.

Wang, Zhengkui, Fan, Qi, Wang, Huiju, Tan, Kian-Lee, Agrawal, Divyakant, and El Abbadi, Amr. Parallel graph olap over large-scale attributed graphs. In *ICDE*, pp. 496–507, 2014. doi: 10.1109/ICDE.2014.6816676. URL http://dx.doi.org/10.1109/ICDE.2014.6816676.

Zheng, Yu, Zhang, Lizhu, Xie, Xing, and Ma, Wei-Ying. Mining interesting locations and travel sequences from gps trajectories. In *WWW*, pp. 791–800. ACM, 2009. ISBN 978-1-60558-487-4. doi: 10.1145/1526709.1526816. URL http://doi.acm.org/10.1145/1526709.1526816.