A Framework for Discovering Frequent Event Graphs from Uncertain Event-based Spatio-temporal Data

Piotr S. Maciąg
Institute of Computer Science, Warsaw University of Technology, Nowowiejska 15/19, 00-665, Warsaw, Poland

Keywords: Uncertain Spatio-temporal Data, Probabilistic Sequential Patterns, Fuzzy Microclustering.

Abstract: The aim of this paper is to discuss a novel framework designed for discovering frequent event graphs from uncertain spatio-temporal data. We consider the problem of discovering hidden relations between event types and their set of uncertain spatio-temporal instances. For that purpose, we designed the following data mining framework: microclustering of uncertain instances, generating set of possible worlds according to the possible worlds semantic technique, creating a microclustering index for each world, generating a set of event graphs from created microclusters and defining apriori based algorithm mining frequent event graphs (EventGraph Miner). To the best of our knowledge this is the first approach to discover hidden patterns from event-type spatio-temporal data when dataset contains uncertain instances. While the paper does not present experimental results for the proposed framework, it presents its potential for futher studies in the topic.

1 INTRODUCTION

Discovering various forms of patterns from spatio-temporal data is gaining attention of researchers nowadays (Li, 2014). This fact is motivated by the rapid development of sensing techniques, designing new types of sensor networks and introducing new types of spatial and temporal data. The classification of spatio-temporal data distinguishes two basic types of such data: event-based and trajectory-based (Li, 2014).

In the article, we refer to the second from mentioned type of data. The notion of sequential pattern discovered from reliable event-based spatio-temporal data has been introduced in (Huang et al., 2008). The task is to discover all significant sequences of event types, where for any two consecutive types contained in the sequence, instances of the preceding event type attract in their spatio-temporal neighborhoods occurrences of instances of the following event type.

The problem of uncertainty is particularly important for spatio-temporal data due to often unreliable nature of sensors gathering physical or environmental signals, noise introduced by transmission protocols or faults in data storage methods. Unfortunately, the problem of discovering probabilistic patterns from such type of data is still not well investigated. On the other hand, the problem of discovering knowledge from several types of uncertain data (e.g. transaction databases) has been considered in the literature ((Chui et al., 2007; Aggarwal et al., 2009; Chui and Kao, 2008; Aggarwal, 2009; Zhang et al., 2008; Bernecker et al., 2009; Leung and MacKinnon, 2014)). An extensive overview of methods for mining uncertain data is given in (Aggarwal, 2009). The problem of discovering probabilistic sequential patterns from uncertain trajectory data is considered in (Li et al., 2013). A solution for querying uncertain trajectory data is given in (Emrich et al., 2012). The problem of discovering various types of patterns from spatial co-location data under uncertainty is considered in (Ouyang et al., 2017; Wang et al., 2016).

The common technique adapted for discovering knowledge from uncertain data is *possible worlds semantic*. The technique is based on generating possible worlds containing certain occurrences of instances and calculating probability of an occurrence of the world. Such probabilities are used in estimating significance of a pattern occurring over multiple worlds.

The notions important for our article are event type and event instances. We assume existence of event types set $F = \{f_1, f_2, \ldots, f_n\}$ and corresponding dataset of event instances D. Each instance $e \in D$ is associated with the following set of attributes: event identifier, event type and a list of possible locations for both spatial and temporal domain, where each location is given with certain probability of occurrence. Additionally, probabilities of occurrences in each lo-

Maciąg, P.
A Framework for Discovering Frequent Event Graphs from Uncertain Event-based Spatio-temporal Data.
DOI: 10.5220/0007411206560663
In *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2019)*, pages 656-663
ISBN: 978-989-758-351-3
Copyright © 2019 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

cations list sum to one. The task considered in the paper is to discover hidden relations between event types based on the set of their uncertain instances. As an attempt to solve the problem we provide a novel framework described in section 2.

In the paper, we introduce the notion of expected frequent event graph and we give a framework for mining such graphs. Frequent event graphs represent a patterns between event types in set F. Designed framework discovers expected frequent event graphs in the following manner: first the set of uncertain instances and their possible locations are microclustered according to their event types. The aim of microclustering is to reduce the size of dataset by merging any locations of the same instance contained in the microcluster. From the set of created microclusters, we generate possible worlds according to the *possible worlds semantic*. Then, for each world and its set of microclusters, we create a microclustering index which contains the following information: identifiers of microclusters in each world, a list of instances contained in each microcluster, an event type representative for the microcluster and centroid location in both spatial and temporal aspects. Based on the set of created microclusters for a particular world we create event graphs. The event graph is a directed acyclic graph which nodes correspond to microclusters and their labels to event types. The edges of the graph are created based on the user given specification of neighborhood spaces between microclusters. The support of an event graph is the number of isomorphic subgraphs of event graphs in a particular world. The expected support of a given event graph is the sum of supports of such graph in all generated worlds weighted by the probabilities of occurrences of these worlds. Frequent event graphs are those with expected support greater than the given threshold.

The following parts of the article are organized as follows. The proposed framework for mining frequent event graphs is given in section 2. We provide basic notions in section 2.1. In section 3, we give an algorithm for microclustering uncertain instances. The steps for generating possible worlds and microclustering index for each world are described in section 3.1. In section 4, our algorithm discovering frequent event graphs is discussed. Section 5 contains conclusions, discussion and possible future work.

2 A FRAMEWORK FOR MINING FREQUENT EVENT GRAPHS

The process of mining expected frequent event graphs is performed according to the framework shown in Figure 1. In the designed framework, first data is compressed and transformed from the set of uncertain event instances to the set of microclusters. In the next step, the set of possible worlds is generated for created microclusters. In section 4, we introduce an event graph for created microclusters. The result of the algorithm is the set of event graphs for which the expected support is greater than the given minimum support threshold (we define such graphs as frequent event graphs).

2.1 Basic Notions

We provide elementary notions important for our formulation of considered problem. The two sets are provided: event types $F = \{f_1, f_2, \ldots, f_N\}$ and dataset of uncertain event instances D.

Definition 1 (Spatio-temporal Space V_{ST})**.** By V_{ST} we denote the whole embedding spatio-temporal space for a given problem.

Let us consider an example given in Figure 2. Note that the spatial area is usually given by two dimensions (i.e. longitude and latitude). For simplicity we denote the spatial aspect in Figure 2 with only one dimension. The number of event types in Figure 2 is 3.

Definition 2 (Uncertain Event Instance)**.** Uncertain event instance $e \in D$ is a triple associated with the following constants: instance ID, event type and a list of possible locations of e, where each location is given with a certain probability of an occurrence. We denote these constants using the following notation: $e.ID$, $e.EventType$, $e.Locations = (l_1, l_2, \ldots, l_n)$. Furthermore, for each possible location $e.l$, $e.l.Loc$ denotes spatial coordinates, $e.l.Time$ occurrence time and $e.l.Probability$ probability of an occurrence.

Let us consider an example given in Figure 2. Instance e_1 is of type A and has two possible locations. The two assumptions are important to note: (1) we assume that each possible location is occurring in a point, (2) probabilities of occurrences of locations are summing to one (that is, having an instance e, $\sum_{l_i \in e.Locations}(e.l_i.Probability) = 1$, where i is indexing locations of an instance e).

The common technique adapted for discovering various types of patterns from uncertain data in data mining is *possible worlds semantic*. In the technique, possible worlds are generated as all possible combinations of occurrences of instances locations: in each word only one location of each uncertain instance may occur. The probability of an occurrence of the world is calculated as the product of probabilities of

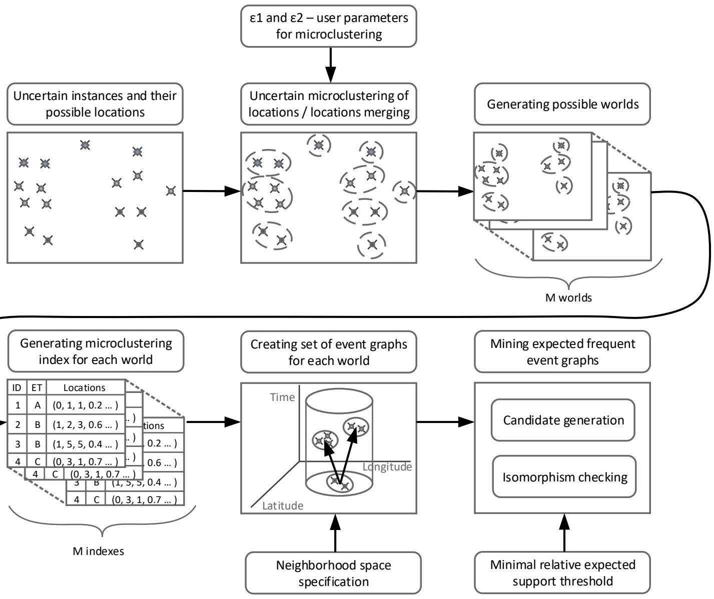

Figure 1: The framework for modelling and mining frequent event graphs.

Table 1: An example of event-based dataset with attributes: instance ID, event type, locations list.

ID	Type	Locations list (spatial coordinate, time, probability)
e_1	A	$l_1 = ((25,3),0.2), l_2 = ((33,2),0.8)$
e_2	A	$l_1 = ((69,4),0.3), l_2 = ((78,2),0.5), l_3 = ((79,4),0.2)$
e_3	B	$l_1 = ((33,5),1.0)$
e_4	B	$l_1 = ((23,5),0.5), l_2 = ((31,6),0.5)$
e_5	B	$l_1 = ((21,6),1.0)$
e_6	B	$l_1 = ((72,7),0.4), l_2 = ((78,7),0.4), l_3 = ((92,6),0.2)$
e_7	C	$l_1 = ((22,8),0.6), l_2 = ((33,8),0.4)$
e_8	C	$l_1 = ((50,9),0.5), l_2 = ((78,9),0.5)$

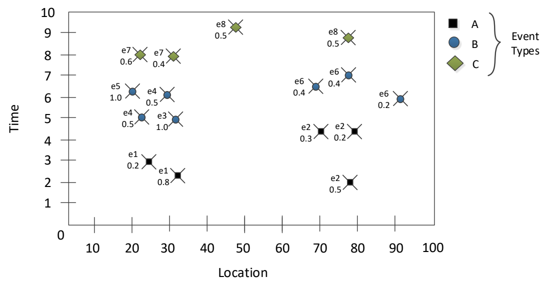

Figure 2: The visualization of the dataset given in Table 1.

occurrences of locations occurring in that world. Having generated world, the support of a pattern discovered in such world is weighted by the probability of an occurrence of that world. The overall estimated support of a pattern in the dataset D is the sum of weighted supports of the pattern occurring in each possible world.

Generating possible worlds directly from the set of instances is in general infeasible. The number of such worlds is bounded by $|D|^{Max}$, where $|D|$ denotes the number of instances in D and Max denotes the maximal number of possible locations among all instances in D. An attempt to solve this problem is to reduce the number of instances locations by performing microclustering of locations.

3 UNCERTAIN MICROCLUSTERING

In this section, we consider the problem of uncertain microclustering of instances locations. The idea behind approach is: (1) to reduced dataset size; (2) proceed with discovering patterns using created microclusters rather than particular instances. The set of possible microclusters and locations contained in them before generating possible worlds for dataset in Figure 2 is given in Figure 4. By $D(f)$ in Algorithm 1, we denote the set of instances of event type f in D.

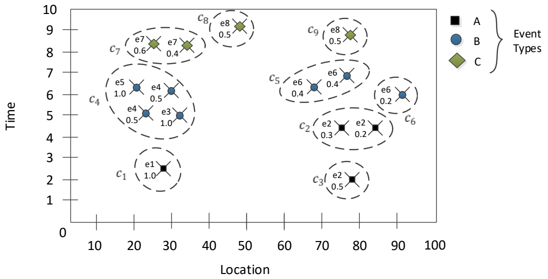

Figure 3: A set of uncertain microclusters for dataset given in Figure 2 before merging locations of the same instance inside microclusters.

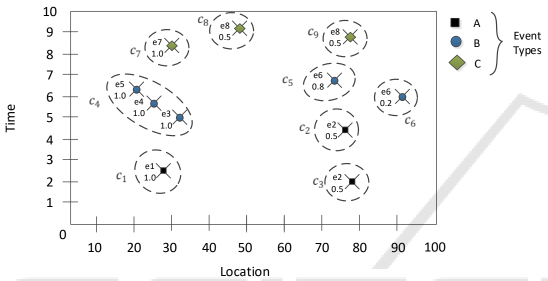

Figure 4: A set of uncertain microclusters for dataset given in Figure 2 after merging locations of the same instance inside microclusters.

In Algorithm 1, ε_1 and ε_2 are the neighborhood distances for the microclustering algorithm. Algorithm 1 iteratively verifies each possible location of each instance and if that location is not in a microcluster, then is inserted to the new microcluster. Algorithm 2 retrieves all possible locations of instances other than $e.l$, which are within distances ε_1 and ε_2 from $e.l$ and are of the same type as $e.EventType$. All these locations are inserted to the same microcluster as $e.l$. ε_1 is the maximal spatial distance between spatial coordinates $e.l.Loc$ and spatial coordinates of any other instance location (say $p.l.Loc$) of type f (that is, $p.l$ is considered to be the spatial neighbor of $e.l$ if $distance(e.l.Loc, p.l.Loc) \leq \varepsilon_1$). Distance function is the Euclidean norm. ε_2 is the maximal temporal interval between $e.l.Time$ and any other possible location of instances of type f. $p.l$ is considered to be a temporal neighbor of $e.l$ if $|e.l.Time - p.l.Time| \leq \varepsilon_2$.

The purpose of step 9 in Algorithm 1 is to merge all locations of the same instance occurring inside a microcluster into one possible location. The probability of such location is the sum of probabilities of merged locations and its spatial coordinates and occurrence time are calculated as averages of spatial coordinates and occurrences times of merged locations.

Table 2: Uncertain microclusters generated for dataset given in Table 1.

MCID	Event Type	Contained instances
c_1	A	$e_1 : 1.0$
c_2	A	$e_2 : 0.5$
c_3	A	$e_2 : 0.5$
c_4	B	$e_3 : 1.0, e_4 : 1.0, e_5 : 1.0$
c_5	B	$e_6 : 0.8$
c_6	B	$e_6 : 0.2$
c_7	C	$e_7 : 1.0$
c_8	C	$e_8 : 0.5$
c_9	C	$e_8 : 0.5$

3.1 Generating Possible Worlds

Having a set of uncertain microclusters the next step is to generate all possible worlds containing certain occurrences of such microclusters. Table 2 contains uncertain microclusters depicted in Figure 4. Instances e_2, e_6 and e_8 may occur in different microclusters and, due to that, there exist 8 possible worlds. The first world contains the following microclusters and instances: $w_1 = \{c_1 = \{\ e_1\}, c_2 = \{e_2\}, c_4 = \{e_3, e_4, e_5\}, c_5 = \{e_6\}, c_7 = \{e_7\}, c_8 = \{e_8\}\}$. The second one is $w_2 = \{c_1 = \{e_1\}, c_3 = \{e_2\}, c_4 = \{e_3, e_4, e_5\}, c_5 = \{e_6\}, c_7 = \{e_7\}, c_8 = \{e_8\}\}$, etc. The probability of an occurrence of each possible world is the product of probabilities of instances locations contained in the microclusters generated for such world and is calculated according to Eq. 1. Having generated M worlds for a given set of uncertain microclusters $\sum_{i=1}^{M}(P(w_i)) = 1$.

$$P(w) = \prod_{c \in w} \prod_{e.l \in c} e.l.Probability \qquad (1)$$

For each possible world and for its microclusters we generate a *microclustering index*. The microclustering index contains identifiers of microclusters, event type contained in a particular microcluster, a list of contained locations of instances and the centroid location for a microcluster. In Table 3, we give a microclustering index for world 1.

4 DISCOVERING FREQUENT EVENT GRAPHS

In Figure 5, we depict world w_1 which contains microclusters and instances locations generated according to method proposed in subsection 3.1.

Definition 3 (Spatio-temporal Neighborhood Space)**.** For a microcluster c, its spatio-temporal neighbor-

Algorithm 1: Algorithm for generating uncertain microclusters.

Input: D - dataset of uncertain instances; $\varepsilon_1, \varepsilon_2$ - neighborhood distances.

for each event type $f \in F$ **do**
    for each instance $e \in D(f)$ **do**
        for each location $l \in e.Locations$ **do**
            if $e.l$ does not belong to any microcluster **then**
                Create new microcluster with MCID := MCID + 1.
                Label $e.l$ as belonging to the microcluster MCID.
                X := RetrieveNeighbors($e.l, \varepsilon_1, \varepsilon_2$)
                Label each location $p.l \in X$ as belonging to the microcluster MCID.
                Merge all locations of the same instance in MCID into one location.
            end if
        end for
    end for
end for

Table 3: The microclustering index created for microclusters of world 1 shown in Figure 5.

MCID	Event type	Contained locations	Num. of locations	Centroid location
c_1	A	$e_1.l_1 = ((28,2),1.0)$	1	(28, 2)
c_2	A	$e_2.l_1 = ((75,4),0.5)$	1	(75, 4)
c_4	B	$e_3.l_1 = ((33,5),1.0)$, $e_4.l_1 = ((27,5),1.0)$, $e_5.l_1 = ((21,6),1.0)$	3	(27, 5)
c_5	B	$e_6.l_1 = ((75,7),0.8)$	1	(75, 7)
c_7	C	$e_7.l_1 = ((27,8),1.0)$	1	(27, 8)
c_8	C	$e_7.l_1 = ((50,9),0.5)$	1	(50, 9)

Algorithm 2: RetrieveNeighbors function.

Input: $e.l$ - a possible location of instance e; $\varepsilon_1, \varepsilon_2$ - neighborhood distances.

Return all possible locations of instances other than $e.l$ within distance ε_1 and ε_2 from $e.l$.

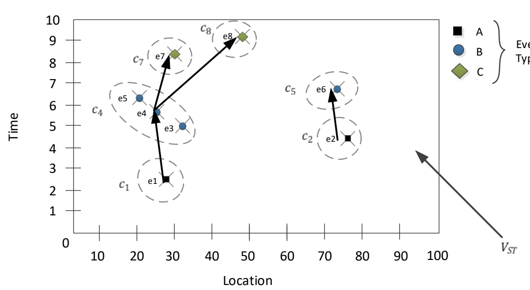

Figure 5: Microclusters from world 1 and *event graphs* created for them.

hood space $\mathcal{V}_c$ is a cylindrical space defined by spatial radius R and temporal interval T. The centroid's spatial coordinates of the microcluster are the center of spatial circle with radius R and temporal interval of length T is defined beginning with the time stamp of centroid.

In Figure 6, we show spatio-temporal cylindrical space for microcluster c_1 with parameters $R = 4, T = 10$.

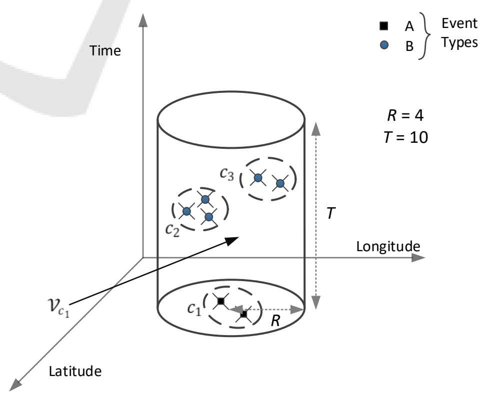

Figure 6: Spatio-temporal neighborhood space of microcluster c_1.

Definition 4. Given a microcluster c, its spatio-temporal neighborhood space $\mathcal{V}_c$, we say all microclusters which centroids are contained inside $\mathcal{V}_c$ follow c (in other words, there is a following relation between such microclusters and c).

For microcluster c_1 and its neighborhood space V_{c_1} shown in Figure 6, microclusters c_2, c_3 of event type B are following c_1.

Definition 5 (Set of Event Graphs $\mathcal{G}$)**.** Given generated world w and a set of microclusters for that world, a set of graphs $\mathcal{G}_w$ contains all directed acyclic graphs with nodes corresponding to microclusters and edges denoting the following relations between microclusters. The graph $G = (V,E,L,\lambda) \in \mathcal{G}_w$ is defined as follows:

1. $V = (v_1, v_2, \ldots, v_n)$ is the set of vertices corresponding to microclusters of a given world.
2. $E = (x_1, x_2, \ldots, x_m)$ is the set of edges denoting the following relation between microclusters.
3. L is the set of event types.
4. $\lambda : V \to L$ is the function assigning to each vertex an event type of a microcluster represented by this vertex.

Example 1**.** In Figure 5, we show microclusters generated for world 1. For this world $\mathcal{G}_{w_1}$ contains two directed acyclic graphs: G_1:

- $V_1 = \{v_1, v_2, v_3, v_4\}$.
- $E_1 = \{(v_1,v_2),(v_2,v_3),(v_2,v_4)\}$.
- $L_1 = \{A,B,C\}$
- $\lambda_1(v_1) = A, \lambda_1(v_2) = B, \lambda_1(v_3) = C, \lambda_1(v_4) = C$.

and G_2:

- $V_2 = \{v_1, v_2\}$.
- $E_2 = \{(v_1,v_2)\}$.
- $L_2 = \{A,B\}$.
- $\lambda_2(v_1) = A, \lambda_2(v_2) = B$.

Definition 6 (Supporting Graph of an Event Graph)**.** Given an event graph G, its supporting graph $G' \in \mathcal{G}_w$ is the graph which subgraph H is isomorphic to G i.e. for the mapping ϕ of vertices between G and H $\phi : V(G) \to V(H)$ the following two conditions are preserved:

1. Any vertices v_1 and v_2 with an edge (v_1,v_2) in G also have an edge $(\phi(v_1),\phi(v_2))$ in H.
2. The event types of v_1 and v_2 are the same as $\phi(v_1)$ and $\phi(v_2)$.

Definition 7 (Support and Relative Support of an Event Graph)**.** For a considered event graph G, the support of G ($sup_w(G)$) is the number of graphs in $\mathcal{G}_w$ supporting G in world w. The relative support of G ($relSup_w(G)$) is defined as $\frac{sup_w(G)}{|\mathcal{G}_w|}$, where $|\mathcal{G}_w|$ is the number of graphs in $\mathcal{G}_w$.

Let us consider an event graph $G = (V = \{v_1,v_2\}, E = \{(v_1,v_2)\}, lbl = \{A,B\}, (L(v_1) = A, L(v_2) = B))$. Its support for $\mathcal{G}$ given in Example 1 is 2 and relative support is 1.

Definition 8 (Expected Support and Relative Expected Support of an Event Graph)**.** Having generated M worlds, the expected support of an event graph G is:

$$expSup(G) = \sum_{i=1}^{M} P(w_i) \cdot sup_{w_i}(G) \tag{2}$$

where $sup_{w_i}(G)$ is support of G in world w_i. The relative expected support of G is defined as:

$$relExpSup(G) = \sum_{i=1}^{M} P(w_i) \cdot \frac{sup_{w_i}(G)}{|\mathcal{G}_{w_i}|} \tag{3}$$

Definition 9 (Problem Definition)**.** Given a dataset of uncertain instances D, the task is to discover all event graphs G with $relExpSup(G) \geq minSup$. All such graphs will be referred as frequent event graphs and denoted by $\mathcal{F}$.

4.1 Algorithm Discovering Frequent Event Graphs

Algorithm 3 is inspired by the apriori-based strategy to mine frequent graphs (Jiang et al., 2013). The cardinality of the graph is the number of its vertices. First, the set $\mathcal{F}_1$ is generated from event graphs sets $\mathcal{G}_{w_1}, \mathcal{G}_{w_2}, \ldots, \mathcal{G}_{w_M}$. Then, iteratively while $\mathcal{F}_{k-1}$ is not empty, the following operations are performed:

- A set of candidate event graphs of cardinality k is generated from $\mathcal{F}_{k-1}$ by procedure Candidate-gen($\mathcal{F}_{k-1}$).
- The support of each graph $G \in C_k$ in a world w_i is calculated as the number of graphs isomorphic with G in $\mathcal{G}_{w_i}$.
- The relative expected support of G is calculated according to Definition 8.
- G is included in $\mathcal{F}_k$ if its relative expected support is greater than *minSup* threshold.

The candidate graphs may be generated according to the procedure presented in (Kuramochi and Karypis, 2004).

5 CONCLUSIONS AND DISCUSSION

In the paper, we proposed a new framework for discovering patterns from event-based spatio-temporal data. The proposed framework consists of the following steps: data microclustering, generating possible worlds from uncertain microclusters and creating microclustering index for each generated world, then for

Algorithm 3: Apriori-based EventGraph Miner.

Input: $\mathcal{G}_{w_1}, \mathcal{G}_{w_2}, \ldots, \mathcal{G}_{w_M}$ - sets of event graphs for worlds $1 \ldots M$; *minSup* - minimal relative expected support threshold.

Output: $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_k$ - sets of frequent event graphs with cardinality 1 to k.

$\mathcal{F}_1 \leftarrow$ sets of frequent event graphs of cardinality 1 detected for $\mathcal{G}_{w_1}, \mathcal{G}_{w_2}, \ldots, \mathcal{G}_{w_M}$.
$\mathcal{F}_2 \leftarrow$ sets of frequent event graphs of cardinality 2 detected for $\mathcal{G}_{w_1}, \mathcal{G}_{w_2}, \ldots, \mathcal{G}_{w_M}$.
while $\mathcal{F}_{k-1} \neq \emptyset$ **do**
  $\mathcal{F}_k \leftarrow \emptyset$.
  $C_k \leftarrow$ Candidate-gen($\mathcal{F}_{k-1}$).
  for each $G \in C_k$ **do**
    relExpSup(G) $\leftarrow 0$.
    for each event graphs set G_{w_i} **do**
      $\text{sup}_{w_i}(G) \leftarrow 0$.
      for each $H \in \mathcal{G}_{w_i}$ **do**
        if Is-isomorphism(G,H) **then**
          $\text{sup}_{w_i}(G) \leftarrow \text{sup}_{w_i}(G) + 1$.
        end if
      end for
      relExpSup(G) $\leftarrow$ relExpSup(G) + $P(w_i) \cdot \frac{\text{sup}_{w_i}(G)}{|\mathcal{G}_{w_i}|}$.
    end for
    if relExpSup(G) $\geq$ minSup **then**
      $\mathcal{F}_k \leftarrow \mathcal{F}_k \cup G$.
    end if
  end for
end while

each world based on its microclustering index generating a set of event graphs and discovering expected frequent event graphs from given dataset. The several points of the proposed framework shall be further discussed:

- The method for microclustering dataset. We proposed rather simple method for dataset microclustering. The more complex approaches may be to apply one of the well known density based algorithms (DBSCAN or OPTICS).
- For generating possible worlds and microclustering set. While the aim of microclustering is to merge location of uncertain instances and reduce the number of generated worlds, the number of generated worlds still may be significant. That can make the algorithm discovering expected frequent event graphs infeasible.

In our future work, we will focus on improving the notions provided in the paper and performing experimental results showing efficiency and effectiveness of the proposed algorithms. Some preliminary experiments performed by us show that the possible bottleneck of the proposed solution is the number of generated possible worlds despite performing microclustering step. In such a case, further improvements of the solution should focus on more efficient generation of possible worlds and calculating support of event graphs.

REFERENCES

Aggarwal, C. C. (2009). *Managing and Mining Uncertain Data*. Springer Publishing Company, Incorporated.

Aggarwal, C. C., Li, Y., Wang, J., and Wang, J. (2009). Frequent pattern mining with uncertain data. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 29–38, New York, NY, USA. ACM.

Bernecker, T., Kriegel, H.-P., Renz, M., Verhein, F., and Zuefle, A. (2009). Probabilistic frequent itemset mining in uncertain databases. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 119–128, New York, NY, USA. ACM.

Chui, C.-K. and Kao, B. (2008). *A Decremental Approach for Mining Frequent Itemsets from Uncertain Data*, pages 64–75. Springer Berlin Heidelberg, Berlin, Heidelberg.

Chui, C.-K., Kao, B., and Hung, E. (2007). *Mining Frequent Itemsets from Uncertain Data*, pages 47–58. Springer Berlin Heidelberg, Berlin, Heidelberg.

Emrich, T., Kriegel, H. P., Mamoulis, N., Renz, M., and Zufle, A. (2012). Querying uncertain spatio-temporal data. In *2012 IEEE 28th International Conference on Data Engineering*, pages 354–365.

Huang, Y., Zhang, L., and Zhang, P. (2008). A framework for mining sequential patterns from spatio-temporal event data sets. *IEEE Transactions on Knowledge and Data Engineering*, 20(4):433–448.

Jiang, C., Coenen, F., and Zito, M. (2013). A survey of frequent subgraph mining algorithms. *The Knowledge Engineering Review*, 28(1):75–105.

Kuramochi, M. and Karypis, G. (2004). An efficient algorithm for discovering frequent subgraphs. *IEEE Transactions on Knowledge and Data Engineering*, 16(9):1038–1051.

Leung, C. K. S. and MacKinnon, R. K. (2014). Fast algorithms for frequent itemset mining from uncertain data. In *2014 IEEE International Conference on Data Mining*, pages 893–898.

Li, Y., Bailey, J., Kulik, L., and Pei, J. (2013). Mining probabilistic frequent spatio-temporal sequential patterns with gap constraints from uncertain databases. In *2013 IEEE 13th International Conference on Data Mining*, pages 448–457.

Li, Z. (2014). *Spatiotemporal Pattern Mining: Algorithms and Applications*, pages 283–306. Springer International Publishing, Cham.

Ouyang, Z., Wang, L., and Wu, P. (2017). Spatial co-location pattern discovery from fuzzy objects. *In-*

ternational Journal on Artificial Intelligence Tools, 26(02):1750003.

Wang, L., Han, J., Chen, H., and Lu, J. (2016). Top-k probabilistic prevalent co-location mining in spatially uncertain data sets. *Frontiers of Computer Science*, 10(3):488–503.

Zhang, Q., Li, F., and Yi, K. (2008). Finding frequent items in probabilistic data. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pages 819–832, New York, NY, USA. ACM.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS