Local rules associated to k-communities in an attributed graph

Henry Soldano and Guillaume Santini and Dominique Bouthinon
Université Paris13, Sorbonne Paris Cité
LIPN, CNRS, (UMR 7030)
F-93430, Villetaneuse, France
{henry.soldano,guillaume.santini,dominique.bouthinon}@lipn.univ-paris13.fr

***Abstract*—We address the problem of finding local patterns and local rules in an attributed graph. A (global) closed pattern is the most specific attribute pattern shared by the vertices of the (possibly simplified) subgraph induced by some attribute pattern. A local closed pattern is the maximal attribute pattern associated to a particular dense region of this subgraph. As such local regions, we are in particular interested in k-communities of pattern subgraphs. In this case we show that there is a closure operator such that, given a pattern q subgraph and a k-community in this subgraph, returns the local closed pattern shared by all the members of the community. We then consider how to generate triples (c, e, l) where c is a (global) closed pattern whose subgraph contains e as a k-community, and l is the corresponding local closed pattern. This leads to implication rules expressing what new attributes are specific of the k-community e in the pattern c subgraph.**

***Keywords*-attributed networks; k-community; closed pattern mining; knowledge discovery**

I. Introduction

We address here the problem of discovering local patterns in an attributed graph. Most previous work focus on the topological structure of the graph, thus ignoring the vertex properties, or consider constrained local or semi-local patterns [1]. In [2] patterns on co-variations between vertex attributes are investigated in which topological attributes are added to the original vertex attributes. In [3] the authors investigate the correlation between the support set of an attribute set and the occurrence of dense subgraphs. In this article, we consider a graph $G = (O, E)$ whose vertices are described in some pattern language L. The patterns occurrences in the vertex set O, i.e. their *support sets*, is then submitted to connectivity constraints that reveals various dense regions in the graph. In the standard closed itemset mining approach developed in Formal concept Analysis (FCA) [4], Galois Analysis [5], and Data Mining (see for instance [6], a support-closed pattern, i.e. a pattern which is maximal, in terms of specificity, within the equivalence class of all patterns sharing the same support set, is the maximum element of its equivalence equivalence class and is easily computed using a closure operator. Furthermore the equivalence classes lead to implication rules that hold on the dataset under investigation.

In a previous work [7] the attributed graph $G = (O, E)$ is investigated in the following way: each pattern support set $e \subseteq O$, as a set of vertices, induces a subgraph $G(e)$ of G, and this subgraph is then simplified by removing vertices in various ways. The vertices of such an *abstract subgraph* all satisfy some topological constraint, as for instance belonging to a k-clique, and form the *abstract support set* of the pattern. What happens here is that the extensional space is then reduced to a part A of 2^O, called a *graph abstraction.* Graph abstractions are defined in such a way that applying a closure operator, we obtain *abstract closed patterns*, i.e. the maximum among patterns sharing the same abstract support set, together with *abstract implication rules* corresponding to inclusion of abstract support sets of patterns q and w and denoted by $\Box^A q \to \Box^A w$. In this article, given some attribute pattern, we are interested in extracting *local support closed patterns*, i.e. maximal attribute patterns each associated to one dense subgraph, so allowing to extract *local implication rules* particular to specific dense groups of objects. Recently the closed pattern mining methodology has been extended to *local closed patterns*: they are obtained by applying a set of *local closure operators* [**?**]. The extensional space of local support sets is then a confluence, i.e. a structure weaker than a lattice. In the graph case, this means that from the support set of some (closed) pattern c, various dense support sets $e_1, \dots e_k$, called *local support sets* and belonging to F are extracted and each associated to a *local support closed pattern* , i.e. the most specific pattern l_i common to the elements of the local support set e_i. Again we obtain a set of *local implication rules* corresponding to inclusion of local support sets, but now such an implication is only valid in the vicinity of some dense group of vertices m, and we write them $\Box^A_m q \to \Box^A_m w$. As we will see below when investigating k-communities, it may be interesting to define *indirect local implication rules* which are local implication rules defined in a new vertex space. All these forms of implication rules are related by an inference order, stating $R \vdash R'$ whenever from validity of rule R we may infer validity of rule R'. For instance, from validity of standard rule $q \rightarrow w$ on a dataset O we may infer, whatever is the abstraction A of 2^O, the validity of the corresponding abstract rule $\Box^A q \rightarrow \Box^A w$. In the same way from a valid global rule (possibly abstract) we may validity of a local rule. In the latter case, let $F \subseteq A$ be a confluence of 2^O, $\Box^A q \rightarrow \Box^A w$ be a valid abstract implication, where the support set of w includes some element m of F, then the local implication $\Box^A_m q \rightarrow \Box^A_m w$ is valid.

First, we describe the case in which the confluence is the set of vertex subsets inducing connected subgraphs of some attributed graph. Figure I, we display a graph whose vertices represents pupils on a school in the West of Scotland, edges represent friendship relations and vertex attributes concern substance use and sporting activity[1]. As a running example we consider the empty pattern whose support set is the whole vertex set O and start from a graph abstraction that deletes from a support set the pupils that do not belong to any friendship triangle in the subgraph induced by the support set of some pattern q (here the empty pattern).The subgraph induced by the remaining (colored or dark) vertices in Figure I is made of 4 connected components. At that point, the corresponding abstract closed pattern reveals what attributes have in common the pupils in this abstract support set. Now, we are interested in knowing whether in each of the two connected components whose size is at least 4 there is some additional attributes shared by their members. The set of attributes shared by all the pupils in such a connected component is a local support closed pattern. However, the largest connected component is clearly made of distinct dense parts, i.e. communities, we would like to consider when defining local closed patterns. For that purpose, we can generalize the local closed approach in order to detect the k-communities (see [8]) of the subgraphs induced by support sets of patterns. Figure I displays the 5 3-communities of size at least 4 found in the whole graph of West Scotland pupils.

Each of these k-community is associated to a local closure of the empty pattern. For instance, the red 3-community $\{1, 10, 11, 14, 15, 16\}$ of the whole graph has $C12, D34m$ as its local closed pattern. Because the global closed pattern here also is the empty pattern (there is no item common to the vertices in all these communities), this leads to the local rule:

$\Box_{1,10,11,14,15,16}\ \{\}\ \rightarrow \Box_{1,10,11,14,15,16}\{C12, D34m\}$
stating that $\{C12, D34m\}$ is a pattern common to all the pupils of this 3-community: in this community the pupils have no cannabis consumption (C12) but a moderate to high alcohol consumption (D34m).This can also be rewritten as

[1]http://www.stats.ox.ac.uk/~snijders/siena/s50_data.htm

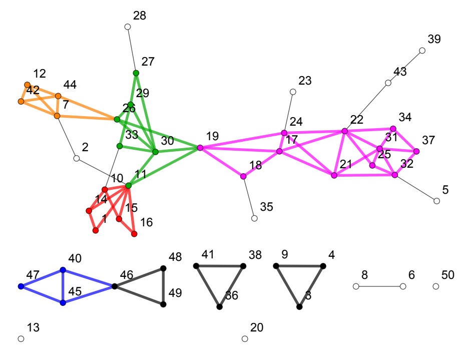

Figure 1. The original friendship graph of a group of West Scotland pupils. When excluding vertices not belonging to any triangle (the empty circles) we found 4 connected components, two of which have size at least 4. The pupils belonging to the same connected components of adjacent triangles in the derived graph share the same color. We only consider here such 3-communities when their size is at least 4.

$\Box_{11,15,16}\ \{\}\ \rightarrow \Box_{11,15,16}\{C12, D34m\}$
meaning that all pupils belonging to the same 3-community as the pupils triangle $11, 15, 16$ in the whole triangle graph share the pattern $\{C12, D34m\}$.

We are interested here in finding a basis of these local implication rules, representing what additional knowledge is provided by looking at implications that hold in the dense vicinity of particular vertices. This leads to the following mining problem:

- Find the set of all k-communities, of size at least s, generated from the subgraph of G induced by the support set of some pattern of L and compute, for each k-community e, the corresponding local closed pattern l together with all local rules $\Box_e c \rightarrow \Box_e l \setminus c$ where c is a closed pattern whose local closure is l.

A direct way to solve the mining problem is a top-down search in the pattern space and consists in extending a separate and conquer algorithm, namely PARAMINER [9], to output with no repetition these (c, e, l) triples. Complete description of the direct algorithm is outside the scope of this article, however we describe the general idea of the algorithm in Section V.

II. Closed patterns, abstractions and confluences

In order to make the article as self-contained as possible, we firs recall definitions and results about closure operators and recall that support closed patterns are obtained as the images of a closure operator. We then discuss abstractions, and abstract support closed patterns, also obtained applying a closure operator. When applied

[1]We note implication rules indifferently $\Box_e c \rightarrow \Box_e l$ or, as in this example, $\Box_e c \rightarrow \Box_e l \setminus c$.

to the set of vertex subsets of some attributed graph, graph abstractions constrain the extensional space to vertex subsets satisfying some property. More recently abstractions have been generalized to weaker structures called confluences. Graph confluences will allow to constrain the extensional support sets to dense parts of the graph.

A. Preliminaries

Definition 1: Let E be an ordered set and $f : E \to E$ a self map such that for any $x, y \in E$, f is monotone, i.e. $x \leq y$ implies $f(x) \leq f(y)$ and idempotent, i.e. $f(f(x)) = f(x)$, then:

- If $f(x) \geq x$, f is called a closure operator
- If $f(x) \leq x$, f is called an interior operator.

In the first case, an element such that $x = f(x)$ is called a closed element.

A well known result on closure operators on lattices has the following dual variant:

Property 1: Let T be a lattice. A subset A of T is the range $p[T]$ of some interior operator on T, if and only if A is closed under join. The interior operator $p : T \to T$ is then defined as $p(x) = \vee_{\{a \in A | a \leq x\}} a$ and A is a lattice. When T is a power set, as 2^O, the meet and joins operator simply are the intersection $\cap$ and union $\cup$ operators. The intuition is that $p(x)$ is the greatest element of A included in x.

In data mining the set of occurrences of a pattern q, belonging to some pattern language L, as 2^I, is known as the *support set* $\text{ext}(q)$ of pattern q and a pattern q is said *support-closed* whenever it is a maximal pattern among those sharing the same support set. Now, whenever there is a unique support closed pattern corresponding to a given support set e, as it is the case in the itemset mining framework, an intension function $\text{int}(e)$ returns this support-closed pattern, and the map $\text{int} \circ \text{ext}$ is a closure operator. This leads to a connection between the two spaces, L and 2^O called a Galois connection, defining closure operators on both space.

Projected or abstract Galois lattices have been recently defined by noticing that applying an interior operator on 2^O [10], [11] we obtain again closure operators:

Property 2: Let X and L be two lattices, (int, ext) be a Galois connection on (X, L) and p be an interior operator on X, and $A = p[X]$ the associated abstraction, we have that $(int, p \circ ext)$ is a Galois connection on (A, L), i.e.:

$f = \text{int} \circ p \circ \text{ext}$ is a closure operator on L,

The abstract support set of pattern q is obtained as $p \circ \text{ext}(q)$. What happens here is that a new equivalence relation is defined such that $q \equiv_A w$ whenever $p \circ \text{ext}(q) = p \circ \text{ext}(w)$, each equivalence class of which corresponds to some element e of A and has a maximum, i.e. a unique *abstract support closed* pattern. Note that, as p is monotone, whenever $\text{ext}(q) \subseteq \text{ext}(w)$, i.e. $q \to w$ is valid we also have $p \circ \text{ext}(q) \subseteq p \circ \text{ext}(w)$, i.e. the abstract implication $\Box^A q \to \Box^A w$ is also valid. We say that from $q \to w$ we infer $\Box^A q \to \Box^A w$.

Now to introduce locality in the closure framework, we have to consider confluences which are structures weaker than lattices investigated in [12] and close to confluent families introduced by Mario Boley and co-authors [13][2]. Extensional confluences restrict the extensional space 2^O or A to a subset F. In this case the support set e of a pattern q is projected, through interior operators, on various smaller and disjoint local support sets $\{e_i\}$. We obtain then local closure operators f_i, each leading to a local closed pattern: $f_i(q)$ is then the most specific pattern shared by objects in the local support set e_i. We define hereunder confluences through a characteristic property:

Property 3: Let X be a lattice and $F \subseteq X$, F is a confluence of X if and only if for any x, y, t in F with $x \geq t$ and $y \geq t$, we have that $x \vee y$ belongs to F.

A confluence is associated to a set of interior operators each defined on an up set $X^t = \{x \in X | x \geq t\}$ of X

Lemma 1: Let F be a confluence of a lattice X,

- the mapping $p_t : X^t \to X^t$ such that $p_t(x) = \vee_{q \in F^t \cap X_x} q$, is an interior operator and $p_t[X^t] = F^t$.
- if $q \leq t$, and $x \in X^t$, then $p_t(x) = p_q(x)$

A consequence of this result is that we only need the minimal elements of F,min$[F]$, to characterize a confluence, i.e. F is a confluence of X if and only if for any x, y in F with $x \geq m$ and $y \geq m$, where m is a minimal element of F, we have that $x \vee y$ belongs to F.

Inclusion of local support sets define local implication rules, denoted $\Box_t^A q \to \Box_t^A w$ and whenever $F \subseteq A$, $\Box^A q \to \Box^A w$ is valid and $t \subseteq \text{ext}(w)$ we also have that $\Box_t^A q \to \Box_t^A w$ is valid.

Of course some valid abstract implications are not valid implications and some valid local implications are not valid abstract implications. As a result abstract and local rules add some new knowledge to what can be directly observed from the support sets of patterns.

III. Local closures and local implications

In the example that follows, we define a graph confluence by considering a non-directed graph.

Example 1: Let $O = \{1, 2, 3, 4\}$, $G = (O, E)$ be a graph whose vertex set is O and edge set is E. Let $F \subseteq 2^O$ be the set of vertex subsets inducing connected subgraphs of G. F is a confluence whose set of minimal elements is $M = \{\{1\}, \{2\}, \{3\}, \{4\}\}$, i.e. the set of singletons of 2^O. The union of two vertex subsets each

[2]The two definitions differs on the following point the empty itemset belongs to any confluent family while a confluence may have several minimal elements.

inducing a connected subgraph of G that contains a given singleton s is a vertex subset obviously inducing a connected subgraph of G: s connects the two subgraphs, and therefore F is a confluence of 2^O. The projection $p_{\{s\}}$ projects then any vertex subset S containing s on the connected component of $G(S)$ containing s. The up set $F^{\{s\}}$ is then the set of vertex subsets inducing connected subgraphs containing s and the union of all these $F^{\{s\}}$ represents the whole set of connected subgraphs of G. For the sake of simplicity we will further write singletons $\{s\}$ as s and subsets as words as for instance 123. The subset $F^{1+3} = F^1 \cup F^3$ representing vertex subsets inducing connected subgraphs containing vertices 1 or 3 also is a confluence. Figure 2 displays the diagram of F^{1+3}.

□

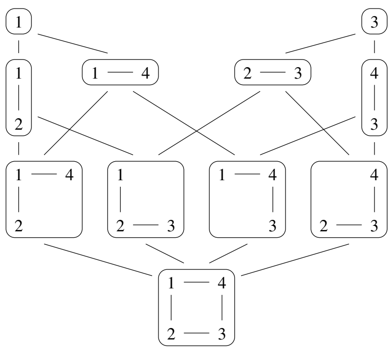

Figure 2. The Hass diagram of a family F of connected subgraphs each generated by a vertex subset of the original graph whose vertex subset is $\{1, 2, 3, 4\}$ and whose edges form the square $\{12, 23, 34, 14\}$. We only display here the part $F^{1+3} = F^1 \cup F^3$ of F, which also has the confluence structure: F^1 and F^3 both are lattices whose join operator is the set theoretic union and therefore F^{1+3} is a confluence.

We have then a result stating that the set $\text{int}[F]$ of intensions of a confluence F, i.e. local support closed patterns, is obtained by joining the ranges of a set of local closure operators. We first need to define these local closure operators:

Property 4: Let e be an element of $X = 2^O$, $L_{int(e)}$ be the down set of L whose maximum is $\text{int}(e)$ and X^e be the upset of X whose minimum is e, F be a confluence on X, e an element of F and p_e the corresponding interior operator on X^e, then $f_e = \text{int} \circ p_e \circ \text{ext}$ is a closure operator on $L_{int(e)}$

f_e is a called a *local closure operator* with respect to e.

When considering a given (possibly abstract) closed pattern c with respect to 2^O, whose local support set e contains m, and whose corresponding local closed pattern in F is l, we have that the implication rule $\Box_m c \rightarrow \Box_m l$ holds. The set of such $\Box_m c \rightarrow \Box_m l$ local implications, with $c \neq l$, represents (a basis for) the local knowledge deriving from the reduction of the extensional space from 2^O to the confluence F.

IV. Graph abstractions and graph confluences

We consider that the set of objects O is the set of vertices of a graph $G = (O, E)$ whose edges represents relation between objects. A vertex is labelled with an element from a language of patterns L. From now on, without loss of generality, we will consider a set of attributes (or items) X and 2^X as the pattern language.

A. Graph abstractions

Following proposition 1 an abstraction $A \subseteq 2^O$ is defined as a part of 2^O closed under union and can equivalently be defined as $p[2^O]$ where p is an interior operator on 2^O. The following Lemma defines a way to build abstractions:

Lemma 2: Let $P : O \times 2^O \rightarrow \{\text{true}, \text{false}\}$ be such that *i)* $P(x, e)$ implies $x \in e$ and *ii)* $e \subseteq e'$ and $P(x, e)$ implies $P(x, e')$, then the iteration of the function q defined as $q(e) = \{x \in e | P(x, e)\}$ reaches a fixed-point and the operator p defined as $p(e) = \text{fixed} - \text{point}(q, e)$ is an interior operator. P is then called the characteristic property of the corresponding abstraction.

A graph abstraction is defined through a characteristic property $P(x, e)$ which expresses some minimal connectivity requirement of the vertex x within the induced subgraph G_e, as for instance the degree $\geq k$-graph abstraction $A_{\text{degree} \geq k}$ that states that a subset of vertices e belongs to $A_{\text{degree} \geq k}$ whenever $d(x) \geq k$ for all x in G_e. We give hereunder two characteristic properties of graph abstractions we are interested in:

1) $cc \geq s$: x has to belong to a connected component of size at least s in G_e
2) k-clique: x belongs to some k-clique of G^e.

It is interesting to note that we can combine two (or more) abstractions A_1 and A_2. For instance, we may consider abstract subgraphs whose vertices both belong to a k-clique and to a connected component exceeding a minimal size s.

B. Graph confluences

A graph confluence is a confluence of 2^O where O is the vertex set. The simplest graph confluence, called a cc-confluence is obtained by considering only vertex subsets inducing connected subgraphs as exemplified above: for any vertex v, and any vertex subset $e \subseteq O$, $p_{\{v\}}(e)$ is the connected component of $G(e)$ that contains v.

In what follows, we will consider a family $T \subseteq 2^O$, and consider T as the vertex set of a *derived* graph $G_T = (E_T, T)$. We consider then the cc-confluence F of 2^T as the extensional space and search for the corresponding local closed patterns. The corresponding local support sets are afterwards transformed into support sets in 2^O. Let $u : 2^T \to 2^O$ be such that $u(e_T) = \cup_{t \in e_T} t$. $u(e_T)$ is called the *flattening* of e_T. We consider then the two maps ext_T and $\mathrm{int_T}$ defined as follows:

- $\mathrm{ext}_T : L \to 2^T$ with $\mathrm{ext}_T(p) = \{t | t \subseteq \mathrm{ext}(p)\}$
- $\mathrm{int}_T : 2^T \to L$ with $\mathrm{int}_T(e_T) = \mathrm{int} \circ u(e_T)$

$\mathrm{ext}_T(p)$ represents the support set of p in T when considering that p occurs in t whenever p occurs in all elements of t as a subset of O. Conversely $\mathrm{int}_T(e_T)$ represents the greatest pattern in L whose support set in T includes e_T, i.e. whose support set in O contains, as subsets, the elements of e_T. We have then the following result when flattening the (local) support sets so found in F:

Property 5: Let F be a confluence of 2^T and $U = u[F]$, where u is the flattening operator on O, then

- $\mathrm{int}_T[F] = \mathrm{int}[U]$
- Let (e_T, l) a local (local support set, local closed pattern) pair and $e_T \geq m \in \min[F]$, then $u(e_T)$ is the greatest element of $u[F^m]$ among elements e such that $\mathrm{int}(e) = l$.

Note that we may as well start from an abstract extension on 2^T deriving from an abstract extension on 2^O and preserve the equivalence between $\mathrm{int}_T[F]$ and $\mathrm{int}[U]$.

Now, a k-community in a graph G is the flattening (in the sens defined above) of a connected component when considering the graph G_T derived by considering the family T of k-cliques of G. Therefore, $\mathrm{int}[U]$ is the set of most specific patterns each occurring in a k-community induced by the support set of some pattern.

Example 2: Let $G = (O, E)$ be the graph displayed on the left part of Figure 3. Each vertex of G belongs to some triangle in G, therefore G is the same as its triangle abstraction. Each vertex has an itemset included in $\{a, b, c\}$ as a label. The set of triangles is $T = \{t_0, t_1, t_2, t_3, t_4, t_5, t_6, t_7\}$ and form a triangle graph G_T displayed on the right part of Figure 3. An edge relates any pair of triangles sharing two vertices in G, as for instance (t_0, t_1). Each triangle in G_T has as its itemset the intersection of the itemsets of its three vertices in G. For instance, the description of t_1 in G_T is $ac = abc \cap ac \cap ac$. The vertex subsets inducing connected subgraphs of G_T form the confluence $F^T = \{\{t_0\}, \{t_1\}, \{t_0, t_1\}, \{t_2\}, \{t_3\}, \{t_2, t_3\}, \{t_4\}, \{t_5\}, \{t_4, t_5\}, \{t_6\}, \{t_7\}, \{t_6, t_7\}\}$. We do not consider in this example the empty pattern.

The support set of the pattern a is $\mathrm{ext}(a) = \{t_0, t_1, t_2, t_3, t_6, t_7\}$. The local support with respect to t_0 is $p_{t_0}(\{t_0, t_1, t_2, t_3\}) = \{t_0, t_1\}$, i.e. the connected component containing $\{t_0\}$ of the subgraph induced by $\mathrm{ext}(a)$.

- $f_0(a) = f_1(a) = ac$, $f_2(a) = f_3(a) = ab$, $f_6(a) = f_7(a) = ab$

In the same way, the pattern b whose support set is $\mathrm{ext}(b) = \{t_2, t_3, t_4, t_5\}$. leads to the following local closed patterns:

- $f_2(b) = f_3(b) = ab$, $f_4(b) = f_5(b) = bc$, $f_6(b) = f_7(b) = ab$

Note that ab appears both as a local closed pattern resulting from a with respect to f_2, f_3 and f_6, f_7, and as a local closed pattern resulting from b with respect to f_2, f_3 and again to f_6, f_7. Now, as both a and b are closed patterns with respect to 2^T, we obtain various triples in the form (a, e_T, l) and (b, e_T, l) corresponding to local implications in the form $\Box_{e_T} a \to \Box_{e_T} l$ and $\Box_{e_T} a \to \Box_{e_T} l$. The former, for instance, also rewrites as $\Box_{\{t_i\}} a \to \Box_{\{t_i\}} l$ where t_i is any element of e_T. This leads to the following sets of local implications:

- $\Box_{\{t_2\}} a \to \Box_{\{t_2\}} ab$, $\Box_{\{t_3\}} a \to \Box_{\{t_3\}} ab$, equivalent to $\Box_{\{t_2, t_3\}} a \to \Box_{\{t_2, t_3\}} ab$
- $\Box_{\{t_6\}} a \to \Box_{\{t_6\}} ab$, $\Box_{\{t_7\}} a \to \Box_{\{t_7\}} ab$,
- $\Box_{\{t_2\}} b \to \Box_{\{t_2\}} ab$, $\Box_{\{t_3\}} b \to \Box_{\{t_3\}} ab$,

$\Box$

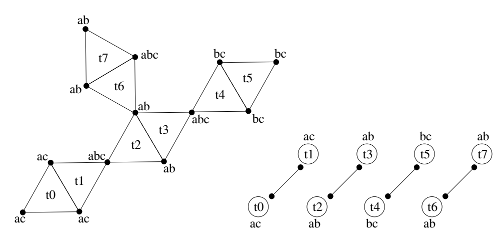

Figure 3. On the left we have a graph of objects each described as an itemset included in $\{a, b, c\}$. This graph represents the triangle abstraction of some input graph. On the right, the graph G_T whose vertices are the triangles of G. The itemset describing a vertex in G_T is the intersection of the itemsets describing the elements of the corresponding triangle in G.

C. Experiments

In our experiments we use the CORON software [14] to compute frequent closed patterns, according to some frequency threshold, then apply a set of PYTHON functions as a post processing[3]. Starting from the set of frequent (possibly abstract) closed patterns C we then compute for each such pattern $c \in C$ the subgraph induced by its (abstract) support set, extract the various connected components $\{e_1, ..., e_k\}$ that are large enough, compute the corresponding local closed patterns

[3]The corresponding software is to be found in https://lipn.univ-paris13.fr/~santini/ .

$\{c_1, ..., c_k\}$ and output the corresponding triples. By applying the $l \neq c$ constraint we select the triples expressing some new local knowledge and the corresponding local implication rule basis. When we are interested in computing k-communities, we start from the k-clique graph abstract closed but we also have to build the k-clique graph G_T, where T is the set of k-cliques in G, and compute the local closures corresponding to the subgraphs of G_T induced by the connected components of our abstract closed patterns, and output the corresponding triples, where the local support sets are flattened to be expressed as subsets of O.

1) Teenage Friends and Lifestyle Study's dataset: The dataset is denoted as *s50-1* and is a standard attributed graph dataset[4]. It represents 148 friendship relations between 50 pupils of a school in the West of Scotland, and labels concern the substance use (tobacco, cannabis and alcohol) and sporting activity. Values of the corresponding variables are ordered. The binarization process consists in defining variables representing the value intervals. T stands for Tobacco consumption and has values 1 (no smoking), 2 (occasional) and 3 (regular). C stands for cannabis consumption and has values 1 (never tries) to 4, D stands for alcohol consumption and has values 1 (does not drink) to 5, and S stands for sporting activity and has two values 1 (occasional) and (2) regular. A binary variable represents an interval, as for instance C23 that has value 1 whenever the value of C is in $[2, 3]$. For sake of simplicity we have merged the two highest values in variables T,C and D. For instance values 4 and 5 in alcohol consumption are merged into a 4m (4 and more) value. The binary attributes allow to represent any interval: for instance D=2 is obtained as {D12,D23m}.

We want to answer to the question: "what knowledge can be extracted when considering groups of pupils sharing some pattern and connected by friendship relationships?". For that purpose, we computed the local abstract closures associated to the cc-confluence representing 3-communities in subgraphs of the triangle graph G_T derived from the original graph and the "support$\geq$ 4" constraint on O.

We recall (see Figure I) that the graph G_T is originally made of 8 connected components which only 5 satisfy the minimal size of 4 in O. From an original set of 166 global closed patterns computed on O, 109 became infrequent when applying the local closure, and 57 leads to one of the 14 frequent local closed patterns found in F.

The local closure of the global pattern $\{S2\}$ leads to 3 local patterns belonging to 3 separate connected components of G_T. The 3 corresponding local abstract patterns are namely $\{S2, C1, T1\}$, $\{S2\}$ and $\{S2, C12, D4m\}$. More formally we have the following local implication rules:

$\Box_{21,25,31,32,34,37}$ {S2} $\rightarrow \Box_{21,25,31,32,34,37}$ {C1, T1} ,
$\Box_{11,19,26,27,29,30,33}$ {S2} $\rightarrow \Box_{11,19,26,27,29,30,33}$ {} ,
$\Box_{10,11,15,16}$ {S2} $\rightarrow \Box_{10,11,15,16}$ {C12, D4m} .

Overall, if a pupil do sports regularly and is member of a 3-community larger than 4 pupils, then

- We learn from the first local implication rule that she also never smokes neither tobacco nor cannabis if she belongs to the $21, 25, 31, 32, 34, 37$ community (a very responsible community),
- We do not learn anything from the second local implication rule, concerning the $11, 19, 26, 27, 29, 30, 33$ community as the local closed pattern is identical to the global from which it derives.
- We learn from the last local implication rule that she also never smoke cannabis or tried it just once but drinks alcohol at least once a week if she belongs to the $10, 11, 15, 16$ community (a less responsible community).

It should be noticed that pupil number 11 belongs to the communities corresponding to the two last implication rules. It illustrates that the knowledge extracted is only indirectly related to individuals: the knowledge is extracted at a coarser level, the level of k-cliques, and a same individual may belong to two k-cliques in two different communities.

V. A direct algorithm to compute the set of (c, e, l) triples

A drawback of the indirect approach used in our experiments is that we need to first apply on the (non abstract) closed pattern a global frequency constraint which may prohibit to address large problems. We propose here a direct approach to solve Problem I mentioned above and output with no repetition the (c, e, l) triples w.r.t. a graph G_T and satisfying a frequency constraint on the object set O. For that purpose, we consider a Divide and Conquer algorithm designed to efficiently compute closed itemsets as described in [13] and adapted in [15]. The general idea is to use a Divide and Conquer strategy in order to avoid outputting and specializing patterns computed earlier during the enumeration. The original algorithm described in [13] is correct whatever is the closure operator, and may the be applied to compute abstract closed pattern, provided that the corresponding interior operator is applied to the support set of the pattern to close before intersecting the descriptions of its objects. The original algorithm was shown as polynomial delay, i.e. the delay between outputting two support closed patterns was polynomial in the dataset size. This basically relies on the fact that each closure computation is polynomial and that whenever a closed pattern is

[4] http://www.stats.ox.ac.uk/~snijders/siena/s50_data.htm

avoided (as previously output) the whole branch in the search tree is pruned.

Regarding local closures computation, we use the same algorithm, except that we compute the local closures of each closed pattern C and prune the branch whether C has no frequent local closed pattern. Otherwise, each frequent local closed pattern L deriving from C is associated to its local support set e, together with C. This algorithm ensures that each triple (C, e, L) is enumerated once. Note that the same local support set e may be enumerated several times, associated with various closed patterns C, C'.. but always with the same local closed pattern L.

VI. Conclusion

We have addressed the question of local knowledge to extract from an attributed network, associating to a pattern local support sets, i.e dense parts of the subgraph induced by its support set, and therefore local closed patterns. We have shown that the set of all k-communities associated to attribute patterns form a set of local support sets, to which are associated local closed patterns and local implication rules. We have experimented these ideas using a post processing of the set of closed patterns found on the dataset of attributed vertices, and then proposed a polynomial delay algorithm to enumerate the set of triples (closed pattern, local support set, local closed pattern) from which a basis of local implication rules is extracted.

The central idea is that the notion of support set, i.e. the set of objects in which some pattern occurs, has to be constrained according to the graph structure. The result is not only a set of abstract and local patterns, but also local knowledge, expressed here as abstract and local implication rules. The article proposes a way to extract all k-communities of all pattern subgraphs of an original graph G, and related local knowledge as local implication rules, by considering a derived graph in which vertices are k-cliques. This can straightforwardly be extended to any derived graph obtained by considering a particular family T of vertex subsets as new vertices of the derived graph. However, the basic idea, i.e. defining communities as dense parts of pattern subgraphs and associating to each a most specific pattern shared by the community, can be extended to any way of defining communities. What proposes the present article is a framework in which the computation of pattern subgraphs communities and related local knowledge is efficient using a generic algorithm.

References

[1] P.-N. Mougel, C. Rigotti, and O. Gandrillon, "Finding collections of k-clique percolated components in attributed graphs," in *16th Pacific-Asia Conference, PAKDD 2012, Kuala Lumpur, Malaysia*, ser. Lecture Notes in Computer Science, vol. 7302. Springer, 2012, pp. 181–192.

[2] A. Bechara Prado, M. Plantevit, C. Robardet, and J.-F. Boulicaut, "Mining Graph Topological Patterns: Finding Co-variations among Vertex Descriptors," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 9, pp. 2090–2104, Sep. 2013.

[3] A. Silva, W. Meira, Jr., and M. J. Zaki, "Mining attribute-structure correlated patterns in large attributed graphs," *Proc. VLDB Endow.*, vol. 5, no. 5, pp. 466–477, Jan. 2012.

[4] B. Ganter and R. Wille, *Formal Concept Analysis: Mathematical Foundations*. Springer Verlag, 1999.

[5] N. Caspard and B. Monjardet, "The lattices of closure systems, closure operators, and implicational systems on a finite set: a survey," *Discrete Appl. Math.*, vol. 127, no. 2, pp. 241–269, 2003.

[6] N. Pasquier, R. Taouil, Y. Bastide, G. Stumme, and L. Lakhal, "Generating a condensed representation for association rules," *Journal Intelligent Information Systems (JIIS)*, vol. 24, no. 1, pp. 29–60, 2005.

[7] H. Soldano and G. Santini, "Graph abstraction for closed pattern mining in attributed network," in *European Conference in Artificial Intelligence (ECAI)*, ser. Frontiers in Artificial Intelligence and Applications, T. Schaub, G. Friedrich, and B. O'Sullivan, Eds., vol. 263. IOS Press, 2014, pp. 849–854.

[8] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society." *Nature*, vol. 435, no. 7043, pp. 814–818, Jun 2005.

[9] B. Negrevergne, A. Termier, M.-C. Rousset, and J.-F. Méhaut, "Paraminer: a generic pattern mining algorithm for multi-core architectures," *Data Mining and Knowledge Discovery*, pp. 1–41, 2013.

[10] N. Pernelle, M.-C. Rousset, H. Soldano, and V. Ventos, "Zoom: a nested Galois lattices-based system for conceptual clustering," *J. of Experimental and Theoretical Artificial Intelligence*, vol. 2/3, no. 14, pp. 157–187, 2002.

[11] H. Soldano and V. Ventos, "Abstract Concept Lattices," in *International Conference on Formal Concept Analysis (ICFCA)*, ser. LNAI, P. Valtchev and R. Jäschke, Eds., vol. 6628. Springer, Heidelberg, 2011, pp. 235–250.

[12] H. Soldano, "Closed patterns and abstraction beyond lattices," in *Formal Concept Analysis 12th International Conference, ICFCA 2014, Cluj-Napoca, Romania, June 10-13*, ser. Lecture Notes in Computer Science, C. V. Glodeanu, M. Kaytoue, and C. Sacarea, Eds., vol. 8478. Springer, 2014, pp. 203–218.

[13] M. Boley, T. Horváth, A. Poigné, and S. Wrobel, "Listing closed sets of strongly accessible set systems with applications to data mining," *Theor. Comput. Sci.*, vol. 411, no. 3, pp. 691–700, 2010.

[14] L. Szathmary and A. Napoli, "Coron: A framework for levelwise itemset mining algorithms," in *Third International Conference on Formal Concept Analysis (ICFCA'05), Lens, France, Supplementary Proceedings*, B. Ganter, R. Godin, and E. M. Nguifo, Eds., 2005, pp. 110–113, supplementary Proceedings.

[15] B. Negrevergne, A. Termier, M.-C. Rousset, and J.-F. Méhaut, "Paraminer: a generic pattern mining algorithm for multi-core architectures," *Data Mining and Knowledge Discovery*, pp. 1–41, 2013. [Online]. Available: http://dx.doi.org/10.1007/s10618-013-0313-2