A Galois connection semantics-based approach for deriving generic bases of association rules

S. Ben Yahia, N. Doggaz Y. Slimani, J. Rezgui
Département des Sciences de l'Informatique
Faculté des Sciences de Tunis
Campus Universitaire, 1060 Tunis, Tunisie.
sadok.benyahia;yahya.slimani;narjes.doggaz@fst.rnu.tn;jihen_rezgui@yahoo.fr

Résumé. L'augmentation vertigineuse de la taille des données (textuelles ou transactionnelles) est un défi constant pour la "scalabilité" des techniques d'extraction des connaissances. Dans ce papier, on présente une approche pour la dérivation des bases génériques de règles associatives. Les principales caractéristiques de cette approches sont les suivantes. D'une part, l'introduction d'une structure de données appelée "Trie-itemset" pour le stockage de la relation en entrée. D'autre part, on utilise une méthode "Diviser pour régner" pour réduire le coût de construction de structures partiellement ordonnées, à partir desquelles les bases génériques de règles sont directement extraites.

1 Introduction

Much research in data mining from large databases has focused on the discovery of association rules [Agrawal et Skirant, 1994, Brin *et al.*, 1997, Manilla *et al.*, 1994]. Association rule generation is achieved from a set F of frequent itemsets in an extraction context $\mathcal{D}$, for a minimal support *minsupp*. An association rule r is a relation between itemsets of the form $r: X \Rightarrow (Y - X)$, in which X and Y are frequent itemsets, and $X \subset Y$. Itemsets X and $(Y - X)$ are called, respectively, *antecedent* and *conclusion* of the rule r. The valid association rules are those of which the measure of confidence $Conf(r) = \frac{support(Y)}{support(X)}$ [1] is greater than or equal to the minimal threshold of confidence, named *minconf*. If $Conf(r) = 1$ then r is called *exact association rule (ER)*, otherwise it is called *approximative association rule (AR)*. Exploiting and visualizing association rules is far from being a trivial task, mostly because of the huge number of potentially interesting rules that can be drawn from a dataset. Various techniques are used to limit the number of reported rules, starting by basic pruning techniques based on thresholds for both the frequency of the represented pattern (called the *support*) and the strength of the dependency between antecedent and conclusion (called the *confidence*). More advanced techniques that produce only a limited number of the entire set of rules rely on closures and Galois connections [Bastide *et al.*, 2000, Stumme *et al.*, 2001, Zaki, 2000], which are in turn derived from Galois lattice theory and formal concept analysis (FCA) [Ganter et Wille, 1999]. Finally, works on FCA have yielded a row of results on compact representations of closed set families, also called *bases*, whose impact on association rule reduction is currently under intensive investigation within the community [Bastide *et al.*, 2000, Stumme *et al.*, 2001].

In this paper, we propose a trie-based new data structure called "**Itemset-trie**" tree. Itemset-trie tree extends the idea claimed by the authors of FPTree [Han *et al.*, 2000] and CATS [Cheung et Zaiane, 2003], aiming to improve storage compression and to allow (closed) frequent pattern mining without "explicit" candidate itemsets generation. Next, we propose an algorithm, falling in the characterization "Divide and Conquer" to extract the frequent closed itemsets with their associated minimal generators. It is noteworthy that the derivation of Luxemburger base is based on the exploration of such closed itemsets organized upon their natural partial order (also called *precedence relation*). That's why we construct on the fly, concurrently with the closed itemsets discovery process, the local "iceberg lattice". Such local ordered sub-structures can be drawn quite naturally in a parallel manner. Then, these ordered sub-structures are parsed to derive, in a straightforward manner, local association generic bases. Finally, local bases are merged to generate the global one. Such process can be recapitulated as follows :

- Construct the Itemset-trie
- Construct the local ordered structures
- Merge the local ordered structures to derive association rule bases

The remainder of the paper is organized as follows : In Section 2, we motivate the choice of the Itemset-trie and present an algorithm for its construction. Section 3 presents an algorithm for the construction of the ordered structures. Section 4 concludes the paper and points out future directions to follow.

2 Itemset-Trie data structure

In the context of mining frequent (closed) patterns in transaction databases or many other kinds of databases, an important number of studies rely on Apriori-like "generate - and-test" approach [Agrawal et Skirant, 1994]. However, this approach suffers from a very expensive candidate set generation step, especially with long patterns or when we lower user-requirements. This drawback is reinforced with tediously repeated disk-stored database scans. To avoid the approach bottleneck, recent studies (e.g, the pioneering work of Han *et al.* and its FP-tree structure [Han *et al.*, 2000]) proposed to adopt an advanced data structure, where the database is compressed in order to achieve pattern mining. The idea behind the compact data structure FP-tree is that when multiple transactions share an identical frequent itemset, they can be merged into one with a registered number of occurrences.

Beside a costly sorting step, the proposed FP-Tree structure is unfortunately not suited for an interactive mining process, in which a user may be interested in varying the support value. In this case, the FP-tree should be rebuilt since its construction is support dependent. Although the work presented in [Cheung et Zaiane, 2003] tackles this insufficiency, the proposed structure called CATS in which a single item is represented in a node. That's why we introduce a, support independent, more compact structure called *Itemset-trie*, in which each node is composed by an itemset. To illustrate this compactness, let us consider the transactions database given by Figure 1. Figure 1(a) depicts the associated FP-Tree, while Figure 1(b) represents the associated Itemset-trie.

Example 1 *Let us consider the transaction database given by Figure 1 (Left). Each node has the following structure : < itemset/support >. Initially, the trie is empty and it is composed by only a root node. We begin by processing the first transaction "acfmp". We derive a node from the root and we add a new node containing the string "acfmp/1". Next, we process the transaction "abcfm". This transaction and the previous one have in common (or prefixed) the {a} item. Hence, the first node is split: we keep the node with "a/2" and two nodes are derived containing respectively "bcfm/1" and "cfmp/1". Processing the third transaction "bf" will lead to the creation of a new node "bf/1", directly derived from the root node, since no items are prefixed in common. The process described below is respected until processing all the transactions.*

3 Derivation of the ordered structures

Due to lack of available space, interested reader for key results from the Galois lattice-based

TID	items
1	acfmp
2	abcfm
3	bf
4	bcp
5	acfmp

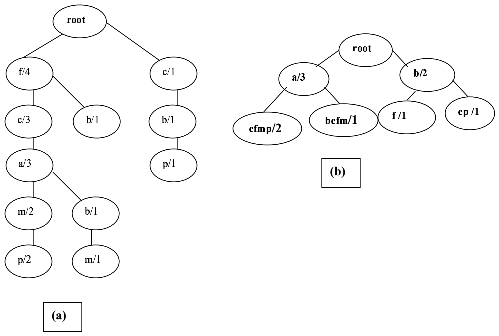

Fig. 1 – ***Left:*** *Transaction database* ***Right:*** *FP-tree and the Itemset-trie associated to the transaction database.*

paradigm in FCA is referred to [Ganter et Wille, 1999].

As output of the first step, we constructed the Itemset-trie. In order to perform an association rule extraction (specially the approximative rules base), we need to construct ordered structures based on the precedence relation.

3.1 Principles

As we work only with closed itemsets, the order construction needs to retrieve the precedence relation from the family of closed itemsets. The main objective (and contribution also) of our approach is to discover the closed itemsets and to order them on the fly. We do not aim to construct only one ordered structure from the input relation (which turns to construct the Hasse diagram), but instead, we look for constructing several ordered structures. Of course, some redundancy will appear i.e. a given closed itemset can appear in more than one ordered structure, but we avoid the expensive cost of Hasse diagram construction. This is performed in a gradual process i.e. by linking one concept at a time to a structure which is only partially finished.

Once the Itemset-trie tree is built, it can be used to mine closed itemsets and their associated minimal generators repeatedly for different support thresholds settings without the need to rebuild the tree. Like FP-growth [Han *et al.*, 2000] and *FELINE*[Cheung et Zaiane, 2003], the proposed algorithm falls in the association rules mining algorithms characterization "Divide and conquer". The initial itemset-trie is fragmented into conditional sub-tries. Indeed, given a pattern called **p**, a p/s conditional itemset-trie tree is built, representing faithfully all transactions that contain pattern p. For example, given the transaction database of Figure 1, the set of 1-itemsets, with their associated supports, is as follows: $< a/3; b/3; c/4; f/4; m/3; p/3 >$. Hence, we have to derive the **a**'s, **b**'s and so on conditional itemset-tries.

It is noteworthy that unlike FP-growth [Han *et al.*, 2000] and Closet [Pei *et al.*, 2002] algorithms, we consider only the lexicographic order and we consider that in a given conditional trie all the remaining 1-itemset should be included. For example, in the above mentioned algorithms (i.e., FP-growth and Closet), the conditional **b**'s trie will not include the 1-itemset {**a**} and that of **c** will exclude both {**a**} and {**b**}. The authors, aiming to discover only closed itemsets, argue that there is no need to include the 1-itemset $\{a\}$ in the b's one, since all closed itemsets containing $\{a\}$ have been already extracted for the a's conditional trie. In our approach, we aim to extract closed itemsets and their associated minimal generators, to construct their associated ordered structure (i.e., Hasse diagram). Since we plan to lead the mining process in a parallel manner, by assigning to each processor a subset of the conditional tries set, each sub-trie should contain an exhaustive description to ensure closed itemsets discovery correctness and to minimize inter-processors communication cost to check itemsets inclusions.

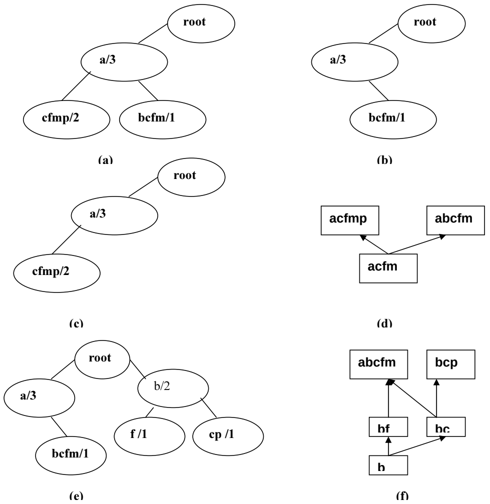

FIG. 2 – *(**a**)$\{a\}$'s conditional Itemset-trie. (**b**)$\{ab\}$'s conditional Itemset-trie.*
*(**c**) $\{ap\}$'s conditional Itemset-trie. (**d**)Ordered structure associated to the 1-itemset $\{a\}$.*
*(**e**)$\{b\}$'s conditional Itemset-trie. (**f**)Ordered structure associated to the 1-itemset $\{b\}$.*

Example 2 *Let us consider the transaction database given by Figure 1(Left). Below, we describe the ordered structures construction for minsup=1. The set of 1-itemsets, with their associated supports, is defined as follows: $< a/3; b/3; c/4; f/4; m/3; p/3 >$. Then starting with the a's conditional Itemset-trie, depicted in Figure 2(a), we can find the associated itemset L_a list:$< b/1; c/3; f/3; m/3; p/2 >$. From such list we remark that 1-itemsets c,f and m are as frequent as the 1-itemset a. Hence, they constitute a closed itemset $\{acfm\}$ with a support equal to 2 and with the 1-itemset $\{a\}$ as its minimal generator. The 1-itemsets c,f and m are removed from L_a. Since it is not empty, we have to go recursively further in depth and to construct the sub-tries, as depicted in Figure 2(b and c), respectively for the 2-itemsets $\{ab\}$ and $\{ap\}$. From L_{ab} we discover the closed itemset $\{abcfm\}$ with a support equal to 1 and with the 2-itemset $\{ab\}$ as its minimal generator. While from L_{ap}, we discover the closed itemset $\{acfmp\}$ with support equal to 1 and with the 2-itemset $\{ap\}$ as its minimal generator. The treatment of L_a ends since there are no more elements to handle. As output, the local Hasse diagram (associated with the a's conditional itemset-trie) can be drawn incrementally. Indeed, the in-depth of L_a list enables to connect, first, the closed itemsets $\{acfm\}$ and $\{abcfm\}$, and second to connect $\{acfm\}$ and $\{acfmp\}$, as depicted in Figure 2(d). The algorithm has to deal next with the L_b list:$< a/1; c/2; f/2; m/1; p/1 >$, extracted from the conditional trie depicted in Figure 2(e). We can check easily that no 1-itemset is so frequent as **b** and then $\{b\}$*

is a closed itemset. Since the remaining list to develop is not empty, we go further in depth and we start with the 2-itemset $\{ab\}$. L_{ab} *is defined as follows:* $< c/1; f/1; m/1 >$ *and from which we discover the closed itemset* $\{abcfm\}$ *with a support equal to 1 and with the 2-itemset* $\{ab\}$ *as its minimal generator. There is no more exploration of this list since it is empty. The closed itemset* $\{b\}$ *is connected to the closed itemset* $\{abcfm\}$*. Next, we have to tackle* L_{bc} *which is equal to* $< a/1; f/1; m/1; p/1 >$*. Any 1-itemset in this list is so frequent as* $\{bc\}$ *and then we can conclude that* $\{bc\}$ *is a closed itemset with a support equal to 2 and having* $\{bc\}$ *as its minimal generator. The list with which to go further in depth remains unchanged. We have respectively to handle* L_{abc}*,* L_{bcf} *and* L_{bcm}*, all yielding the closed itemset* $\{abcfm\}$*. The closed itemset* $\{bc\}$ *is connected to that of* $\{abcfm\}$*. Next, we have to connect* $\{b\}$ *to* $\{bc\}$*. This is performed after a systematic check whether they share a common immediate successor, which is the case in this example. In fact,* $\{bc\}$ *and* $\{b\}$ *are connected respectively to their immediate successor which is* $\{abcfm\}$*. That's why we have to delete the link between* $\{b\}$ *and* $\{abcfm\}$*. The processing of the* L_{bc} *list ends by launching the* L_{bcp} *list, which gives the closed itemset* $\{bcp\}$ *with a support equal to 1 and with* $\{bcp\}$ *as its minimal generator. The associated ordered structure is depicted in Figure 2(f).*

3.2 Derivation of generic bases of association rules

The problem of the relevance and usefulness of extracted association rules is of primary importance. Indeed, in most real life databases, thousands and even millions of high-confidence rules are generated among which many are redundant. This problem encouraged the development of tools for rule classification according to their properties, for rule selection according to user-defined criteria, and for rule visualization. With respect to [Luxemburger, 1991] and [Guigues et Duquenne, 1986], we consider that given a local ordered structure, representing precedence-based relation ordered closed itemsets, generic bases of association rules can be derived in a straightforward manner. In this structure each closed itemset is "decorated" with its associated list of minimal generators. Indeed, AR represent "inter-node" implications, assorted with a statistical information, i.e., the confidence, from a sub-closed-itemset to a super-closed-itemset while starting from a given node in an ordered structure. Inversely, ER are "intra-node" implications extracted from each node in the ordered structure.

4 Conclusion

We presented in this paper a new data structure to extract frequent closed itemsets in order to generate generic bases of association rules. The main characteristics of this structure are. First a compact representation, since in our approach the node represents an itemset while in other approaches, such as FP-Growth and CATS, a node represents only a single attribute. Second, a suited for a "Divide and Conquer" closed itemsets extraction approach. Then, we proposed an algorithm to construct local ordered structures from which it is possible to derive generic bases of association rules. Now, the proposed approach is under experimentation. In the near future, we plan to examine the potential benefits from implementing the proposed approach on an MIMD machine (IBM SP2). Indeed, the construction method leads to a natural parallelization, in the sense that each processor of a parallel architecture can construct locally its ordered structure. Once the local structures are constructed, a master processor can merge them to derive a set of generic bases of association rules.

Références

[Agrawal et Skirant, 1994] R. Agrawal et R. Skirant. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Databases*, pages 478–499, June 1994.

[Bastide *et al.*, 2000] Y. Bastide, N. Pasquier, R. Taouil, L. Lakhal, et G. Stumme. Mining minimal non-redundant association rules using frequent closed itemsets. In *Proceedings of the International Conference DOOD'2000, Lecture Notes in Computer Sciences, Springer-verlag*, pages 972–986, july 2000.

[Brin *et al.*, 1997] S. Brin, R. Motwani, et J. Ullman. Dynamic itemset counting and implication rules. In *Proceedings ACM SIGMOD, International conference on Management of Data ,Tucson, Arizona, USA*, pages 255–264, 1997.

[Cheung et Zaiane, 2003] W. Cheung et O.R. Zaiane. Incremental mining of frequent patterns without candidate generation or support constraint. In *Proceedings of the Seventh International Database Engineering and Applications Symposium (IDEAS 2003), Hong Kong, China*, 16–18,July 2003.

[Ganter et Wille, 1999] B. Ganter et R. Wille. *Formal Concept Analysis.* Springer-Verlag, Heidelberg, 1999.

[Guigues et Duquenne, 1986] J.L. Guigues et V. Duquenne. Familles minimales d'implications informatives résultant d'un tableau de données binaires. *Mathématiques et Sciences Humaines*, (95):5–18, 1986.

[Han *et al.*, 2000] J. Han, J. Pei, et Y. Yin. Mining frequent patterns without candidate generation. In *Proceedings of the ACM-SIGMOD Intl. Conference on Management of Data (SIGMOD'00),Dallas, Texas*, pages 1–12, May 2000.

[Luxemburger, 1991] M. Luxemburger. Implication partielles dans un contexte. *Mathématiques et Sciences Humaines*, 29(113):35–55, 1991.

[Manilla *et al.*, 1994] H. Manilla, H. Toinoven, et I. Verkamo. Efficient algorithms for discovering association rules. In *AAAI Worshop on Knowledge Discovery in Databases*, pages 181–192, July 1994.

[Pei *et al.*, 2002] J. Pei, J. Han, R. Mao, S. Nishio, S. tang, et D. Yang. Closet: An efficient algorithm for mining frequent closed itemsets. In *Proceedings of the ACM SIGMOD DMKD'00, Dallas,TX*, pages 21–30, 2002.

[Stumme *et al.*, 2001] G. Stumme, R. Taouil, Y. Bastide, N. Pasquier, et L. Lakhal. Intelligent structuring and reducing of association rules with formal concept analysis. In *Proceedings of KI'2001 conference, Lecture Notes in Artificial Intelligence 2174, Springer-verlag*, pages 335–350, september 2001.

[Zaki, 2000] M. J. Zaki. Generating non-redundant association rules. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,Boston, MA*, pages 34–43, August 2000.

Summary. The steady growth in the size of data (textual or transactional) is a key progress-driver for more acute knowledge extraction techniques, whose effectiveness and efficiency are constantly challenged. In this paper, we present an approach for deriving generic bases of association rules. The proposed approach is a Galois connection semantics-based. The main features of our approach are: first, to avoid intensive I/O operations, we introduce an advanced trie-based data structure to store the input relation. Second, we use a "Divide and Conquer" method to reduce the construction cost of small partially ordered sub-structures, from which we derive in a straightforward manner association generic bases.