Mining Interesting Disjunctive Association Rules from Unfrequent Items

Ines Hilali[1,2], Tao-Yuan Jen[1], Dominique Laurent[1(✉)], Claudia Marinica[1], and Sadok Ben Yahia[2]

[1] ETIS Laboratory - ENSEA / UCP / CNRS, Cergy-Pontoise, France
[2] Faculty of Sciences of Tunis, University of Tunis El Manar, Tunis, Tunisia
{dlaurent,jen,claudia.marinica}@u-cergy.fr,
ines.hilali@gmail.com, sadok.benyahia@fst.rnu.tn

Abstract. In most approaches to mining association rules, interestingness relies on *frequent* items, i.e., rules are built using items that frequently occur in the transactions. However, in many cases, data sets contain unfrequent items that can reveal useful knowledge that most standard algorithms fail to mine. For example, if items are products, it might be that each of the products p_1 and p_2 does not sell very well (i.e., none of them appears frequently in the transactions) but, that selling products p_1 *or* p_2 is frequent (i.e., transactions containing p_1 *or* p_2 are frequent). Then, assuming that p_1 and p_2 are similar enough with respect to a given similarity measure, the set $\{p_1, p_2\}$ can be considered for mining relevant rules of the form $\{p_1, p_2\} \rightarrow \{p_3, p_4\}$ (assuming that p_3 and p_4 are unfrequent similar products such that $\{p_3, p_4\}$ is frequent), meaning that most of customers buying p_1 or p_2, also buy p_3 or p_4. The goal of our work is to mine association rules of the form $D_1 \rightarrow D_2$ such that (i) D_1 and D_2 are disjoint homogeneous frequent itemsets made up with unfrequent items, and (ii) the support and the confidence of the rule are respectively greater than or equal to given thresholds. The main contributions of this paper towards this goal are to set the formal definitions, properties and algorithms for mining such rules.

Keywords: Data mining · Association rules · Unfrequent items · Similarity measures

1 Introduction

The extraction of association rules is a widely used technique in data mining since it meets the needs of experts in several application fields. Thereby, several studies have focused on *frequent* itemsets mining, i.e., rules are built using items that frequently occur in the transactions. Nevertheless, the application of these patterns is not so attractive in many applications, e.g., intrusion detection, fraud detection, identification of extreme values in data bases, analysis of criminal data, analysis of the genetic confusion from biological data, to cite a few [3,7,10,15].

A. Kawtrakul et al. (Eds.): ISIP 2013, CCIS 421, pp. 84–99, 2014.
DOI: 10.1007/978-3-319-08732-0_7,

Indeed, in such situations, a frequent behaviour may not be of an added value for the end user. However, unfrequent events may be more interesting since they may indicate that an unexpected event or exception has occurred. Thus, the analysis has to be carried out in order to study the possible causes of this unusual deviation from normal behaviour. In this respect, unfrequent (or rare) pattern mining is proved to be of real added value [10]. In fact, rare patterns can identify unusual, unexpected and hidden events [2], since they have a very low frequency in the database.

To illustrate such a statement and standing within the market basket analysis, it is common that each of the products p_1 and p_2 does not sell well (i.e., taken alone, none of them appears frequently in the transactions) but, that selling products p_1 *or* p_2 is frequent (i.e., transactions containing p_1 or p_2 are frequent). Additionally, assuming that a similarity measure between products is provided and that products p_1 and p_2 are similar enough, then the set $\{p_1, p_2\}$ can be considered for mining relevant rules of the form $\{p_1, p_2\} \rightarrow \{p_3, p_4\}$ (assuming that p_3 and p_4 are also unfrequent similar products such that $\{p_3, p_4\}$ is frequent). Such a rule shows that most of customers buying p_1 or p_2, also buy p_3 or p_4. In this rule, $\{p_1, p_2\}$ and $\{p_3, p_4\}$ are seen as two different *homogeneous* frequent sets of products.

It is important to mention that, to the best of our knowledge, no previous work has paid attention to mining association rules in which *unfrequent* items are used to build up itemsets meant to be frequent. To address this issue, we measure the frequency of itemsets according to their *disjunctive* support measure [8]. More precisely, we call disjunctive support of an itemset I, or d-support of I for short, the ratio of the number of transactions containing *at least one* element of I over the total number of transactions. Then, I is said to be *disjunctive-frequent* (or d-frequent, for short) if its d-support is greater than or equal to a fixed threshold. It is important to note that, since any super set of a d-frequent itemset is d-frequent as well, we restrict the set of mined d-frequent itemsets to be *minimal* with respect to set inclusion.

Additionally, another worth of mention feature of our approach is our consideration of "homogeneous itemsets". To define this notion, we assume that a similarity measure between items is given, and then, an itemset I is said to be *homogeneous* whenever all possible pairs of items in I have a similarity degree greater than or equal to a given threshold. Thus, the homogeneity can be seen as a semantic interestingness criterion for selecting relevant itemsets, as done in [12]. Indeed, since in our approach, itemsets are assessed through their disjunctive support, an itemset $\{i_1, i_2\}$ is seen as a generalization of i_1 and i_2, in the sense that, based on the definition of the d-support, this set represents a frequent category of items encompassing i_1 and i_2. Therefore, considering the homogeneity avoids the pitfall of considering heterogeneous itemsets, whose "disjunctive semantics" would then be counter intuitive.

In this context, the association rules that we are interested in are of the form $D_1 \rightarrow D_2$ where D_1 and D_2 are disjoint homogeneous and d-frequent itemsets.

Consequently, we redefine the classical support and confidence measures, respectively called d-support and d-confidence, as follows:

- The d-support of a rule $D_1 \rightarrow D_2$ is the number of transactions containing at least one item in D_1 and at least one item in D_2 over the total number of transactions.
- The d-confidence of a rule $D_1 \rightarrow D_2$ is the ratio of the d-support of $D_1 \rightarrow D_2$ over the d-support of D_1.

However, compared to the standard approach to mining association rules, new issues arise when considering d-support and d-confidence. In fact, it turns out that having at hand the d-supports of D_1 and D_2 does *not* imply that the exact retrieval of the d-support as well as the d-confidence of $D_1 \rightarrow D_2$. Thus, assessing the rules in our approach requires to access the data set. Furthermore, owe to the fulfilment of the monotonicity property of the d-frequent itemsets, we focus on minimal itemsets (with respect to set inclusion), in order to produce only rules whose left- and right- hand sides are minimal (with respect to set inclusion).

To sum up, the main contributions of the present paper are twofold: First, we reconsider our previous work in [8] within the context of transactional databases and we provide the necessary definitions and properties used to show the soundness of mining homogeneous association rules built up with unfrequent items, and using a level wise based exploration algorithm. Second, we provide the associated algorithms for each of the following two steps:

1. Mine *minimal and homogeneous d-frequent itemsets*, referred to as MHDIs in what follows.
2. Use the MHDIs to build up and assess association the rules of interest, which are shown to be of the form $D_1 \rightarrow D_2$ where D_1 is an MHDI, D_2 is a homogeneous d-frequent itemset (not necessarily minimal) disjoint from D_1, and whose d-support and d-confidence are above the given thresholds.

The remainder of the paper is organized as follows: In Sect. 2, we give all basic definitions and properties necessary to state and prove the correctness of our algorithms given in Sect. 3. In Sect. 4, we review several approaches dealing with mining techniques using unfrequent items and we compare these approaches with our work. In Sect. 5, we briefly recall our contributions and we sketch several issues for future work.

2 Formalism and Basic Properties

In this section we give the necessary definitions and properties on which our algorithms rely. We define the notions of disjunctive support of an itemset and of a rule, the disjunctive confidence of a rule, and what we call an homogeneous itemset. These definitions are then used in some basic properties that are necessary to show the correctness of the algorithms to be given in the next section.

2.1 Support and Confidence

We assume a set $\mathcal{I}$ of items that occur in a transaction table Δ whose rows are called transactions. A transaction is a pair (TID, I) where TID is a transaction identifier and I a subset of $\mathcal{I}$, also called an itemset. We borrow from [8] the notion of disjunctive support of an itemset D, that we define as follows.

Definition 1. *For every itemset* D, the disjunctive support of D, or d-support of D *for short, denoted by* $d\text{-}sup(D)$, *is the ratio*

$$d\text{-}sup(D) = \frac{|\{(\text{TID}, I) \in \Delta \mid I \cap D \neq \emptyset\}|}{|\Delta|}.$$

Given a support threshold σ, D *is said to be* disjunctive-frequent, *or* d-frequent *for short, if* $d\text{-}sup(D) \geq \sigma$.

We emphasize that Definition 1 implies that the notion of d-support differs from that of support as defined in [1]. Indeed, given an itemset I, the support of I is computed based on the number of transactions containing *all* items in I.

To illustrate our approach, we consider the following example that will be used as a running example throughout the paper.

Example 1. Let $\mathcal{I} = \{bergerac, cheverny, montlouis, milk, scallop, oyster, salad\}$ be a set of items where *bergerac*, *cheverny* and *montlouis* are names of French wines. We assume the set of transactions Δ as shown in Table 1. To simplify, for every $j = 1, \ldots, 7$, the transaction with TID equal to j is denoted by t_j; for example, t_1 refers to the first transaction in Δ, that is $(1, \{bergerac, milk, scallop\})$.

Table 1. The set of transactions Δ of the running example.

TID	I
1	Bergerac, milk, scallop
2	Cheverny, milk, scallop
3	Scallop
4	Bergerac, milk, oyster
5	Montlouis, oysyer
6	Salad
7	Montlouis

Denoting by $sup(D)$ the support of an itemset D as defined in [1], for $D = \{cheverny, milk\}$ and $D' = \{oyster, milk\}$, we have:

- $sup(D) = \frac{|\{t_2\}|}{7} = \frac{1}{7} = 14.3\,\%$ and $d\text{-}sup(D) = \frac{|\{t_1,t_2,t_4\}|}{7} = \frac{3}{7} = 42\,\%$
- $sup(D') = \frac{|\{t_4\}|}{7} = \frac{1}{7} = 14.3\,\%$ and $d\text{-}sup(D') = \frac{|\{t_1,t_2,t_4,t_5\}|}{7} = \frac{4}{7} = 57.1\,\%$

For a threshold $\sigma = 50\%$, D and D' are not frequent, D is not d-frequent and D' is d-frequent. We also note that no item is frequent with respect to σ. □

It is easy to see that the disjunctive support measure is monotonic with respect to set inclusion, in other words the following proposition holds.

Proposition 1. *For all itemsets D_1 and D_2, if $D_1 \subseteq D_2$ then d-sup$(D_1) \leq$ d-sup(D_2).*

Proposition 1 implies that if $D_1 \subseteq D_2$ and if D_2 is *not* d-frequent, then D_1 can not be d-frequent. Hence, (i) minimal d-frequent itemsets can be mined using a level wise algorithm such as Apriori, and (ii) knowing the minimal d-frequent itemsets allows for knowing *all* d-frequent itemsets (but not their d-supports). We now define disjunctive support and confidence of association rules.

Definition 2. Let D_1 *and* D_2 *be two itemsets. The* disjunctive support, *or* d-support *for short, of* $D_1 \rightarrow D_2$, *denoted by* d-sup$(D_1 \rightarrow D_2)$, *is the ratio*

$$\text{d-sup}(D_1 \rightarrow D_2) = \frac{|\{(\text{TID}, I) \in \Delta \mid (I \cap D_1 \neq \emptyset) \wedge (I \cap D_2 \neq \emptyset)\}|}{|\Delta|}.$$

Given a support threshold σ, $D_1 \rightarrow D_2$ is said to be disjunctive-frequent, *or* d-frequent *for short, if* d-sup$(D_1 \rightarrow D_2) \geq \sigma$.

The disjunctive confidence, *or* d-confidence *for short, of* $D_1 \rightarrow D_2$, *denoted by* d-conf$(D_1 \rightarrow D_2)$, *is the ratio*

$$\text{d-conf}(D_1 \rightarrow D_2) = \frac{\text{d-sup}(D_1 \rightarrow D_2)}{\text{d-sup}(D_1)}.$$

Example 2. In the context of Example 1, for $D_1 = \{bergerac, montlouis\}$ and $D_2 = \{scallop, oyster\}$, we have that:

- d-sup$(D_1) = \frac{|\{t_1,t_4,t_5,t_7\}|}{7} = 57.1\,\%$, d-sup$(D_2) = \frac{|\{t_1,t_2,t_3,t_4,t_5\}|}{7} = 71.4\,\%$,
- d-sup$(D_1 \rightarrow D_2)$ = d-sup$(D_2 \rightarrow D_1) = \frac{|\{t_1,t_4,t_5\}|}{7} = 42.8\,\%$.

As a consequence, for a support threshold $\sigma = 50\,\%$, the two rules $D_1 \rightarrow D_2$ and $D_2 \rightarrow D_1$ are not frequent. On the other hand, we have d-conf$(D_1 \rightarrow D_2) = \frac{42.8}{57.1} = 75\,\%$ and d-conf$(D_2 \rightarrow D_1) = \frac{42.8}{71.4} = 60\,\%$. □

It is important to note from Definition 2 that the notions of d-support and d-confidence of an association rule carry similar semantics as standard support and confidence for association rules. Indeed:

- The d-support of $D_1 \rightarrow D_2$ is the probability that a transaction contains at least one item in D_1 and at least one item in D_2 (recalling that the standard support of $D_1 \rightarrow D_2$ can be seen as the probability that a transaction contains all items in D_1 and all items in D_2).

– The d-confidence of $D_1 \rightarrow D_2$ is the conditional probability that a transaction contains at least one item in D_1 and at least one item in D_2, knowing that it contains at least one item in D_1 (recalling that the standard confidence of $D_1 \rightarrow D_2$ can be seen as the conditional probability that a transaction contains all items in D_1 and all items in D_2 knowing that it contains all items in D_1).

The following proposition states basic properties of d-support and d-confidence.

Proposition 2. *For all itemsets D_1, D_2 and D, we have:*

1. $d\text{-}sup(D_1 \rightarrow D_2) \leq d\text{-}sup(D_j)$ *for* $j = 1, 2$
2. $d\text{-}sup(D_1 \rightarrow D_2) \leq d\text{-}sup((D_1 \cup D) \rightarrow D_2)$
3. $d\text{-}sup(D_1 \rightarrow D_2) \leq d\text{-}sup(D_1 \rightarrow (D_2 \cup D))$
4. $d\text{-}conf(D_1 \rightarrow D_2) \leq d\text{-}conf(D_1 \rightarrow (D_2 \cup D))$
5. $d\text{-}sup(D_1 \rightarrow D_2) = d\text{-}sup(D_1 \rightarrow (D_2 \cup D)) \iff d\text{-}conf(D_1 \rightarrow D_2) = d\text{-}conf(D_1 \rightarrow (D_2 \cup D))$

2.2 Homogeneous Itemsets

As already mentioned, interestingness of itemsets is measured based not only on their d-frequency, but also on their *homogeneity*. Homogeneity of itemsets is defined using a *similarity measure*, that we denote by *sim*. We recall in this respect that similarity measures have been considered in data mining since they allow taking into account semantic aspects in the processing ([12,13,16]).

In this work, we consider the similarity measure defined in [16], called *Total Relatedness*. Assuming a taxonomy over the items, this measure is composed of two other partial similarity measures, called *Highest-Level Relatedness* and *Node Separation Relatedness*, and defined as follows, where i and i' are distinct items in $\mathcal{I}$:

– The Highest-Level Relatedness of i and i', denoted by $HR(i, i')$, is the level of the highest-level node of the path in the taxonomy connecting i and i'.
– The Node Separation Relatedness of i and i' $(i \neq i')$, denoted by $NSR(i, i')$, is the number of nodes in the path connecting i and i' in the taxonomy.

If k is the depth of the taxonomy, the *Total Relatedness measure* is defined by:

$$sim(i, i') = \frac{1 + HR(i, i')}{k * NSR(i, i')}.$$

Example 3. In the context of Example 1, a taxonomy over the items in $\mathcal{I}$ is shown in Fig. 1. In this setting, we have:

– $HR(bergerac, montlouis) = 1$ and $HR(milk, scallop) = 0$,
– $NSR(bergerac, montlouis) = 1$ and $NSR(milk, scallop) = 3$.

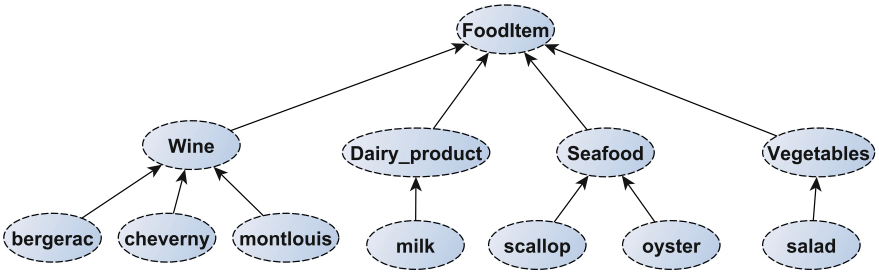

Fig. 1. Taxonomical organization of the items of Example 1

Therefore, according to the definition of *sim* given above, we have: $sim(bergerac, montlouis) = \frac{2}{2} = 1$ and $sim(milk, scallop) = \frac{1}{6}$. □

The notion of homogeneous itemset is defined as follows.

Definition 3. *Let h be a value in the range of the similarity measure sim. An itemset I is said to be* homogeneous *with respect to h if* $\min_{i,i' \in I}(sim(i, i')) \geq h$.

Referring back to Example 3 and considering a similarity threshold $h = 1$, $\{bergerac, montlouis\}$ is homogeneous, whereas $\{milk, scallop\}$ is not.

Since for all itemsets I_1 and I_2 such that $I_1 \subseteq I_2$, $\min_{i,i' \in I_1}(sim(i, i')) \geq \min_{i,i' \in I_2}(sim(i, i'))$ holds, it is easy to see that the following holds.

Proposition 3. *For all itemsets I_1 and I_2 such that $I_1 \subseteq I_2$, if I_2 is homogeneous then I_1 is homogeneous as well.*

In what follows, we call MHDI any *minimal homogeneous d-frequent itemset*, and we consider the problem of mining *all* MHDIs from Δ. An important consequence of Propositions 1 and 3 is that MHDIs can be mined using a level wise algorithm such as Apriori [1].

As another remark concerning homogeneous itemsets, deciding whether an itemset I is homogeneous or not clearly requires a number of tests in $O(|I|^2)$. However, if I is known to be homogeneous, checking whether $I \cup \{i\}$ (where i is an item not in I) is homogeneous only requires to check whether $\min_{i' \in I}(sim(i, i'))$ is greater than or equal to h. Therefore, in this case, checking whether $I \cup \{i\}$ is homogeneous is in $O(|I|)$.

The previous remark is of particular interest in the forthcoming Algorithm 2, where we assume that, for every MHDI D, the set $H(D)$ of all unfrequent items i such that $D \cup \{i\}$ is homogeneous has been computed beforehand.

3 Algorithms

In this section, we present the two algorithms that implement our approach: the first algorithm allows for the computation of all MHDIs whereas the second one allows for the computation of all interesting rules. Before going into the details of these algorithms, we note the following points regarding computations:

1. To avoid computation redundancies when generating candidates, we assume that a total ordering $\prec_{\mathcal{I}}$ over $\mathcal{I}$ is given and that the items of all itemsets are listed according to this ordering.
2. To check whether an itemset I is homogeneous, we assume that all similarity degrees between unfrequent items have been computed and stored. Although we do not provide details regarding this point, we shall see that it can be achieved during the processing of the first algorithm given below.
3. As earlier mentioned, we assume that for every MHDI D, the set $H(D)$ of all unfrequent items i such that $D \cup \{i\}$ is homogeneous has been computed and stored.

3.1 MHDI Computation

As argued in the previous section, MHDIs are mined using a level wise algorithm similar to Apriori [1]. This task is achieved by Algorithm 1 where the set of candidates, denoted by *hom_cand*, is generated by joining the elements of $unfreq\_hom_{k_1}$, i.e., the non d-frequent homogeneous itemsets of size $k-1$ and by pruning the obtained set of itemsets. We note that this assumes that the items of itemsets are listed according to the ordering $\prec_{\mathcal{I}}$.

The main difference with the standard algorithm Apriori is that the homogeneity criterion has to be taken into account, which is achieved line 11. Indeed, if all subsets of $D_1 \cup D_2$ of cardinality $k-1$ are homogeneous, then by Proposition 3, so is $\{i_1^{k-1}, i_2^{k-1}\}$. Therefore, for all i_1 and i_2 in $D_1 \cup D_2$, we have $sim(i_1, i_2) \geq h$, meaning that $D_1 \cup D_2$ is homogeneous.

The correctness of Algorithm 1 can be shown as in [8], where this was done in the context of relational disjunctive queries with no homogeneity criterion. We simply note that, in our context, candidates are selected for the next step if they are non d-frequent and homogeneous. Thus, every non selected itemset is d-frequent *or* not homogeneous. Since the itemsets in *hom_cand* are homogeneous, non selected itemsets for the next step are homogeneous and d-frequent. Moreover, these itemsets are also minimal with respect to set inclusion since their subsets have been previously considered as non d-frequent itemsets. Consequently Algorithm 1 correctly computes all MHDIs.

We now argue that the complexity of Algorithm 1 is similar to that of the standard Apriori algorithm [1]. To see this, we express the complexity of Algorithm 1 in terms of the number of scans of the data set Δ, as for the standard Apriori algorithm. In this case, since Algorithm 1 performs a scan of Δ at each level of the lattice built up with all unfrequent items (see lines 16–19), we obtain that the complexity of Algorithm 1 is linear in the number of unfrequent items (in the same way as the complexity of the standard Apriori algorithm is shown to be linear in the number of frequent items). Hence, similarly to the standard Apriori algorithm, the complexity of Algorithm 1 is in $O(|\mathcal{I}|)$ (considering the worst case when no item is frequent).

Regarding this complexity result, we emphasize that restricting the mined itemsets to be homogeneous has no impact on the complexity of Algorithm 1. We note in this respect that in Algorithm 1 some details have been omitted for

Algorithm 1. Computation of MHDIs

Input: Database Δ, the d-support threshold σ, the similarity threshold h
Output: The set $MHDI(\Delta)$ of all MHDIs

1: // Scan Δ to compute the set of all unfrequent items
2: $unfreq\_hom_1 = \{i \in \mathcal{I} \mid d\text{-}sup(i) < \sigma\}$
3: $MHDI(\Delta) = \emptyset$
4: $k = 2$
5: **while** $unfreq\_hom_{k-1} \neq \emptyset$ **do**
6: $hom\_cand = \emptyset$
7: // Candidate generation
8: **for all** D_1 and D_2 in $unfreq\_hom_{k-1}$ **do**
9: // $D_1 = \{i_1^1, \ldots, i_1^{k-1}\}, D_2 = \{i_2^1, \ldots, i_2^{k-1}\}$
10: **if** $i_1^1 = i_2^1$ and $\ldots$ and $i_1^{k-2} = i_2^{k-2}$ and $i_1^{k-1} \prec_{\mathcal{I}} i_2^{k-1}$ **then**
11: **if** all subsets of $(D_1 \cup D_2)$ of cardinality $k-1$ are in $unfreq\_hom_{k-1}$ **then**
12: $hom\_cand = hom\_cand \cup \{(D1 \cup D_2)\}$
13: **for all** D in hom_cand **do**
14: $d\text{-}sup(D) = 0$
15: // Scan of Δ to compute the d-supports of all D in hom_cand
16: **for all** (TID, I) in Δ **do**
17: **for all** D in hom_cand **do**
18: **if** $D \cap I \neq \emptyset$ **then**
19: $d\text{-}sup(D) = d\text{-}sup(D) + 1$
20: $unfreq\_hom_k = \{D \in hom\_cand \mid d\text{-}sup(D) < \sigma\}$
21: $MHDI(\Delta) = MHDI(\Delta) \cup \{D \in hom\_cand \mid d\text{-}sup(D) \geq \sigma\}$
22: $k = k + 1$
23: **return** $MHDI(\Delta)$

the sake of simplification. In particular, the test line 11 is not necessary when $k = 2$ (because it is always satisfied in this case). However, for $k = 2$, when considering a candidate itemset $D = \{i_1^1, i_2^1\}$, checking whether D is homogeneous amounts to check whether $sim(i_1^1, i_2^1) \geq h$, which in turn, requires to compute $sim(i_1^1, i_2^1)$. This computation does not impact the complexity result mentioned above because similarity is computed based on the given ontology and similarity measure, but *not* based on the data set Δ. We also mention that these similarity results are assumed to be stored in a matrix so as to efficiently compute the sets $H(D)$ for every MHDI D, and other similarity measures needed when running Algorithm 2 to be given next.

The following example illustrates how Algorithm 1 works in the context of our running example.

Example 4. Referring to the data set Δ shown in Table 1, and considering the support threshold $\sigma = 50\%$ and the similarity threshold $h = 1$, Algorithm 1 performs the following steps:

- First, the computation of $unfreq\_hom_1$ line 2 returns $\mathcal{I}$, because, as mentioned in Example 1, no item in $\mathcal{I}$ is frequent.
- For $k = 2$, all possible pairs of items are considered in the loop lines 8–12, and for all distinct i, i' in $\mathcal{I}$, $sim(i, i')$ is computed and stored. Moreover, the

d-supports of the homogeneous pairs are computed through the scan lines 16–19: we obtain that the homogeneous d-frequent itemsets of cardinality 2 are $\{bergerac, montlouis\}$ and $\{scallop, oyster\}$.
Therefore $unfreq\_hom_2$ contains the two itemsets: $\{bergerac, cheverny\}$ and $\{cheverny, montlouis\}$ (since the other unfrequent itemsets of cardinality 2 are not homogeneous).

– For $k = 3$, no candidates are generated line 10, and so the main iteration lines 5–22 stops when k is set to 4, line 22.

Hence, $\{bergerac, montlouis\}$ and $\{scallop, oyster\}$ are the two MHDIs returned by Algorithm 1. □

3.2 From MHDIs to Interesting Association Rules

The output of Algorithm 1 is used to build and assess candidate rules in order to produce the final result. However, in our approach, and contrary to the standard case, the assessment of these candidate rules requires to scan the dataset. This is so because, knowing the d-supports of D_1 and D_2 does not imply that $d\text{-}sup(D_1 \rightarrow D_2)$ can be computed without scanning the data.

Moreover, according to Proposition 2(1), the d-support of a rule $D_1 \rightarrow D_2$ is less than the d-supports of D_1 and D_2. Hence, considering only rules $D_1 \rightarrow D_2$ where D_1 and D_2 are MHDIs is likely to produce a very limited number of rules. On the other hand, Proposition 2(1-2) states that the d-support of a rule $D_1 \rightarrow D_2$ increases when one of the itemsets D_1 or D_2 is enlarged. This is why we look for association rules of the form $D_1 \rightarrow D_2$ where D_1 and D_2 are homogeneous d-frequent itemsets, that might not be MHDIs. Moreover, we naturally require that these sets be "as small as possible" because it is well known that in practice, rules with too many members in their left- and/or right-hand sides are difficult to understand by users.

We now emphasize that, by Proposition 2(2-4), enlarging the right hand side of a rule increases the d-support *and* the d-confidence of the rule, whereas enlarging the left hand side increases the d-support, but not always the d-confidence. In fact, we claim that enlarging the left hand sides of rules is not relevant in our approach. To see this, denoting by $T(D)$ be the set of transactions TID,I in Δ such that $(I \cap D) \neq \emptyset$, we notice that a rule $D_1 \rightarrow D_2$ whose confidence is 1, that is for which $d\text{-}sup(D_1 \rightarrow D_2) = d\text{-}sup(D_1)$, satisfies $T(D_1) \subseteq T(D_2)$. Thus, improving the d-confidence of a rule $D_1 \rightarrow D_2$ whose d-confidence is not 1 tends to make $T(D_1)$ a subset of $T(D_2)$. This can not be achieved by enlarging D_1 because enlarging D_1 entails that $T(D_1)$ is also enlarged.

Based on the previous remarks, the rules we are looking for are of the form $D_1 \rightarrow D_2$, such that, given thresholds σ, γ and h:

1. D_1 is an MHDI and D_2 is a homogeneous d-frequent itemset;
2. D_1 and D_2 are disjoint;
3. $d\text{-}sup(D_1 \rightarrow D_2) \geq \sigma$ and $d\text{-}conf(D_1 \rightarrow D_2) \geq \gamma$;
4. for every rule $D_1 \rightarrow D_2$ satisfying the three items above, and for every $D \subset D_2$, the rule $D_1 \rightarrow D$ does *not* satisfy all three items above.

Algorithm 2. The computation of all interesting association rules

Input: Δ, the set $MHDI(\Delta)$ of all MHDIs, the d-support threshold σ, the d-confidence threshold γ, the similarity threshold h

Output: The set *Result* of all interesting association rules

// **Step 1 :** level $k = 0$

$C = \emptyset$
for all (D_1, D_2) in $MHDI(\Delta) \times MHDI(\Delta)$ **do**
  if $D_1 \cap D_2 = \emptyset$ **then**
    $C = C \cup \{(D_1, D_2, \emptyset, d\text{-}sup(D_1), 0, 0)\}$
Result $= \emptyset$
Scan Δ to compute S-new = $d\text{-}sup(D_1 \rightarrow D_2)$ of rules in C
for all $c = (D_1, D_2, \emptyset, s, 0,$ S-new$)$ in C **do**
  if S-new $\geq \sigma$ and $d\text{-}conf(D_1 \rightarrow D_2) \geq \gamma$ **then**
    $Result = Result \cup \{D_1 \rightarrow D_2\}$
    $C = C \setminus \{c\}$
  else
    if S-new = S-old **then**
      $C = C \setminus \{c\}$
C-old $= C$

// **Step 2:** levels k with $k > 0$

while C-old $\neq \emptyset$ **do**
  C-new $= \emptyset$
  for all $c = (D_1, D_2, E, s,$ S-old, S-new$)$ in C-old **do**
    for all i in $(H(D_2) \setminus (D_2 \cup E \cup D_1))$, and $\max_{\mathcal{I}}(E) \prec_{\mathcal{I}} i$ **do**
      if $(D_2 \cup E \cup \{i\})$ is homogeneous **then**
        $c' = (D_1, D_2, E \cup \{i\}, s,$ S-new$, 0)$
        C-new = C-new $\cup \{c'\}$
  Scan Δ to compute S-new = $d\text{-}sup(D_1 \rightarrow D_2 \cup E \cup \{i\})$ of rules in C-new
  for all $c = (D_1, D_2, E \cup \{i\}, s,$ S-old, S-new$)$ in C-new **do**
    if S-new $\geq \sigma$ and $d\text{-}conf(D_1 \rightarrow D_2 \cup E \cup \{i\}) \geq \gamma$ **then**
      $Result = Result \cup \{D_1 \rightarrow D_2 \cup E \cup \{i\}\}$
      C-new = C-new $\setminus \{c\}$
    else
      if S-old = S-new **then**
        C-new = C-new $\setminus \{c\}$
  C-old = C-new
Delete from *Result* all rules $D_1 \rightarrow D_2 \cup E$ such that *Result* contains $D_1 \rightarrow D_2' \cup E'$ with $D_2' \cup E' \subset D_2 \cup E$
return *Result*

Calling these rules *interesting association rules*, we provide next an algorithm for mining them. Before doing so, we give examples of interesting rules.

Example 5. In the context of our running example, we recall from Example 4 that we have $MHDI(\Delta) = \{D_1, D_2\}$ where $D_1 = \{bergerac, montlouis\}$ and $D_2 = \{scallop, oyster\}$. Thus, the only rules to be considered first are $D_1 \rightarrow D_2$

and $D_2 \rightarrow D_1$. However, as seen in Example 2, none of these rules is d-frequent, since their d-support has been shown to be less than 50 %. Consequently, these rules cannot be interesting.

As suggested just above, in order to get interesting rules we extend the right-hand sides of $D_1 \rightarrow D_2$ and $D_2 \rightarrow D_1$, in order to satisfy the four previous conditions for a confidence threshold $\gamma = 75\,\%$.

- Regarding the rule $D_1 \rightarrow D_2$, we have to extend D_2 into a homogeneous d-frequent itemset D_2', which is not possible because for every item i in $\mathcal{I}$ but not in $D_1 \cup D_2$, $(D_2 \cup \{i\})$ is not homogeneous.
- Considering now $D_2 \rightarrow D_1$, we notice that the item *cheverny* is the only item in $\mathcal{I}$ such that $D_1' = (D_1 \cup \{cheverny\})$ is homogeneous. Moreover:
  - $d\text{-}sup(D_2 \rightarrow D_1') = \frac{|\{t_1,t_2,t_4,t_5\}|}{7} = 57.1\,\%$, and
  - $d\text{-}conf(D_2 \rightarrow D_1') = \frac{57.1}{71.4} = 80\,\%$.

  Therefore $D_2 \rightarrow D_1'$ satisfies the first three conditions above, and as it can be seen that this rule also satisfies the last condition, $D_2 \rightarrow D_1'$ is an interesting association rule. □

3.3 An Algorithm for Mining all Interesting Association Rules

Interesting association rules are mined according to Algorithm 2 in which a 6-tuple $(D_1, D_2, E, s, \text{S-old}, \text{S-new})$ represents the rule $D_1 \rightarrow (D_2 \cup E)$ where:

- D_1 and D_2 are MHDIs, and E is an itemset containing unfrequent items,
- s is the d-support of D_1 (s is known from the run of Algorithm 1),
- S-old = 0 if $E = \emptyset$, and otherwise, S-old is the d-support of a rule of the form $D_1 \rightarrow (D_2 \cup (E \setminus \{i\}))$ from the previous iteration,
- S-new is the d-support of $D_1 \rightarrow (D_2 \cup E)$.

We now discuss the steps of Algorithm 2. In Step 1, we first filter out all pairs of MHDIs (D_1, D_2) where D_1 and D_2 are not disjoint, and the 6-tuples of all remaining potentially interesting rules $D_1 \rightarrow D_2$ are put in the set of candidates (line 4). The supports of these rules are then computed through a scan of Δ (line 6). All candidates whose d-support and d-confidence are respectively greater than or equal to their corresponding threshold are added to the result set *Result* (line 9) and will not be considered in the next step, due to our minimal requirement (line 10). For all other candidates, the test line 12 discards the rule if its d-support is 0, since this means that no transaction d-supports the rule.

The iteration in Step 2 enlarges the right hand sides of the rules in a level wise manner as follows: new candidates are generated by adding one item to the set E thus producing the candidate rule $D_1 \rightarrow (D_2 \cup E \cup \{i\})$ from the rule $D_1 \rightarrow (D_2 \cup E)$. We note that in order to avoid computational redundancies, the ordering $\prec_{\mathcal{I}}$ is used (see line 18). We also emphasize here that we assume that for every D in $MHDI(\Delta)$, the set $H(D)$ has been already computed. Under this hypothesis, line 18, the item i is chosen in $H(D_2)$, because otherwise, $D_2 \cup E \cup \{i\}$ cannot be homogeneous. However, it is still necessary to check whether $D_2 \cup E \cup \{i\}$ is homogeneous, which is done line 19.

Lines 18 to 30 show a processing similar to that of lines 3 to 14 of Step 1, that is: new candidates are generated (lines 17–21) and Δ is scanned in order to compute the current d-supports stored in S-new (line 22). Then, these new candidates are processed according to the fact that they represent or not an interesting rule (lines 23–29). We notice that candidates such that S-new = S-old (line 28) are discarded because Proposition 2(5) shows that, in this case, adding i to the right hand side does not change the d-support and the d-confidence. Therefore, the rule $D_1 \rightarrow (D_2 \cup E \cup \{i\})$ has not to be considered in the next iterations. On the other hand, minimality of the right hand sides of rules is guaranteed line 31, where non minimal rules are discarded. As a consequence, it turns out that Algorithm 2 computes the expected set of all interesting association rules.

We now turn to the study of the complexity of Algorithm 2, which we express in terms of the number of scans of the data set, as done for Algorithm 1. As previously mentioned, in Algorithm 2 candidate rules are generated in a level wise manner by adding one item to the right hand sides of rules, and at each level, the supports and confidences of these candidate rules are computed through one scan of Δ. Therefore, the complexity of Algorithm 2 is the same as that of Algorithm 1, that is in $O(|\mathcal{I}|)$. We also note that, since only items i in $H(D_2)$ are considered to enlarge $D_2 \cup E$, the test of homogeneity line 19, which uses the similarity matrix constructed in Algorithm 1 (for $k = 2$ as previously explained), is linear in the size of E. In other words, the complexity of this test is linear in less than the size of $\mathcal{I}$, even in the worst case.

4 Related Work

Whereas most approaches in data mining are interested in extracting frequent patterns, mining unfrequent or rare patterns (association rules or itemsets) has attracted research efforts these last years [10]. According to these work, rare patterns do not often occur, and are relevant if their elements are strongly correlated. Mining abnormal symptoms in medical applications is a standard example of such patterns. However, it is important to note that, whatever the data set in which these patterns are mined, considering these low-support and high-confidence rules raises major difficulties when using standard association rule mining approaches.

In order to address this issue, some approaches propose to consider the frequency as a relative measure rather than an absolute one, since the items differ from one to another by nature. For example, buying a luxe item is an action much less frequent than buying milk, and so, the corresponding frequencies cannot be interpreted in the same way. In [11], the MSapriori Algorithm has been introduced to mine the absolute unfrequent items by assigning different minimum support thresholds to different items. In [17,20], instead of using different thresholds, a weighted support measurement is used to offer different viewpoints to items. Hence users can assign weights according to their need and find valuable unfrequent patterns. One critical point when applying the previous methods is to assign adequate minimum support thresholds and/or weights to different

items. This task becomes even unfeasible when considering a large number of items. This is why, the relative support measure was proposed in [24].

Although relevant patterns can be mined using these approaches, it turns out that in most cases, very large numbers of candidate itemsets are generated, as in the standard case when the thresholds are set to be very low. Several propositions have been introduced to tackle this issue. The proposition in [18,19] is to find rules directly form their confidence. Although, confidence does not have a downward closure property, the authors use a confidence-based pruning in their rule generation. In this approach, high-confidence and low-support association rules of the form $I \rightarrow i$, where I is an itemset and i is an item, are mined without generating unnecessary low-support itemsets. Other work propose to mine highly correlated patterns using appropriate measures, such as h-confidence [21,22] or Bond [4,23]; relationships between these measures are studied in [14].

It is important to note that most approaches to mining rare association rules concentrate on *conjunctive* patterns built up using unfrequent or frequent items. One main risk when considering conjunctions of unfrequent items is that these itemsets have a very low support, and thus, it is difficult to distinguished such patterns from noisy data. This issue has been investigated in [9] through the notion of exception rule.

As opposed to these approaches, our work is based on *disjunction* to build frequent itemsets, and considers a similarity measure as an additional criterion. We note that it has been shown in [4,23] that the disjunctive form of patterns can be derived from the correlated patterns, using the technique in [5]. However, since the correlated patterns are generated from frequent items, we can not use these results in our approach, where we consider unfrequent items.

We notice that in [6], the authors consider a taxonomy to mine frequent itemsets built up with items from the same level in the taxonomy. However, our approach basically differs from this work because in [6], only frequent items are considered (whereas we consider unfrequent items) and the support threshold is changed according to the level in the taxonomy (whereas we do not change the support threshold during the mining process).

Moreover, the frequent disjunctive itemsets that are mined in our work are built up according to the given taxonomy, but may be different from the concepts defined by this taxonomy. Indeed, in our approach, the taxonomy is used to assess the homogeneity of disjunctive itemsets, according to a given similarity threshold. As a consequence, it is possible that a given homogeneous frequent disjunctive itemset does not "match" existing concepts in the taxonomy, although representing a relevant set of items. We argue that such itemsets can be used to reorganize the ontology, according to the content of the data set and to the similarity threshold chosen by the user (this issue will be investigated in our future work). Hence, it should be clear that the way we use the taxonomy in our approach is radically different from the one in [6].

5 Concluding Remarks

In this paper, we have proposed an approach to mine association rule involving unfrequent items. Unfrequent items are grouped in itemsets to produce frequent itemsets according to the *disjunctive* support measure. In order to produce rules as "understandable" as possible, disjunctive frequent itemsets have been restricted to be minimal with respect to set inclusion, and a homogeneity criterion has been considered for itemsets. We have shown that in this setting, disjunctive frequent itemsets can be mined using a standard level wise algorithm. However, it has also been argued that computing interesting rules in this approach requires further scans of the data set. These scans have been shown to be processed in a level wise manner so as to produce all interesting rules.

We are currently implementing our algorithms to assess their efficiency and their relevancy. To this end, we intend to consider synthetic and real data sets, so as to provide experiments as complete as possible. Regarding further research issues in the context of this work, we mention that considering homogeneous itemsets allows for further investigation. Indeed, homogeneity allows to define groups of unfrequent items, seen as concepts among which rules are to be mined. It seems that the interesting rules considered in this paper are closely related to these rules between concepts. We plan to investigate this issue in the future.

References

1. Agrawal, R., Mannila, H., Srikant, R., Toivonen, R., Verkamo, A.I.: Fast discovery of association rules. In: Advances in Knowledge Discovery and Data Mining, pp. 309–328. AAAI-MIT Press (1996)
2. Berberidis, C., Vlahavas, I.P.: Detection and prediction of rare events in transaction databases. Int. J. Artif. Intell. Tools **16**(5), 829–848 (2007)
3. Booker, Q.E.: Improving identity resolution in criminal justice data: an application of NORA and SUDA. J. Inform. Assur. Secur. **4**, 403–411 (2009)
4. Bouasker, S., Hamrouni, T., Ben Yahia, S.: New exact concise representation of rare correlated patterns: application to intrusion detection. In: Tan, P.-N., Chawla, S., Ho, C.K., Bailey, J. (eds.) PAKDD 2012, Part II. LNCS, vol. 7302, pp. 61–72. Springer, Heidelberg (2012)
5. Hamrouni, T., Ben Yahia, S.: Generalization of association rules through disjunction. Ann. Math. Artif. Intell. **59**(2), 201–222 (2010)
6. Han, J., Fu, Y.: Discovery of multiple-level association rules from large databases. In: PVLDB, pp. 420–431 (1995)
7. He, Z., Xu, X.: FP-Outlier: frequent pattern based outlier detection. Comput. Sci. Inf. Syst. **2**(1), 103–118 (2005)
8. Hilali-Jaghdam, I., Jen, T.-Y., Laurent, D., Ben Yahia, S.: Mining frequent disjunctive selection queries. In: Hameurlain, A., Liddle, S.W., Schewe, K.-D., Zhou, X. (eds.) DEXA 2011, Part II. LNCS, vol. 6861, pp. 90–96. Springer, Heidelberg (2011)
9. Hussain, F., Liu, H., Suzuki, E., Lu, H.: Exception rule mining with a relative interestingness measure. In: Terano, T., Liu, H., Chen, A.L.P. (eds.) PAKDD 2000. LNCS, vol. 1805, pp. 86–97. Springer, Heidelberg (2000)

10. Koh, Y.S., Roundtree, N.: Rare Association Rule Mining and Knowledge Discovery: Technologies for Infrequent and Critical Event Detection. IGI Global, Hershey (2010)
11. Liu, B., Hsu, W., Ma, Y.: Mining association rules with multiple minimum supports. In: ACM International Conference on Knowledge Discovery and Data Mining, SIGKDD, pp. 337–341. ACM (1999)
12. Marinica, C., Guillet, F.: Knowledge-based interactive postmining of association rules using ontologies. IEEE Trans. Knowl. Data Eng. **22**(6), 784–797 (2010)
13. Natarajan, R., Shekar, B.: A relatedness-based data-driven approach to determination of interestingness of association rules. In: ACM Symposium on Applied Computing (SAC), pp. 551–552. ACM (2005)
14. Omiecinski, E.R.: Alternative interest measures for mining associations in databases. IEEE Trans. Knowl. Data Eng. **15**(1), 57–69 (2003)
15. Romero, C., Romero, J.R., Luna, J.M., Ventura, S.: Mining rare association rules from e-learning data. In: Proceedings of the 3rd International Conference on Educational Data Mining (EDM 2010), Pittsburgh, PA, USA, pp. 171–180 (2010)
16. Shekar, B., Natarajan, R.: A framework for evaluating knowledge-based interestingness of association rules. Fuzzy Optim. Decis. Making **3**, 157–185 (2004)
17. Tao, F., Murtagh, F., Farid, M.: Weighted association rule mining using weighted support and significance framework. In: ACM International Conference on Knowledge Discovery and Data Mining, SIGKDD, pp. 661–666. ACM (2003)
18. Wang, K., He, Y., Cheung, D.M.: Mining confident rules without support requirement. In: ACM International Conference on Information and Knowledge Management, CIKM, pp. 89–96. ACM (2001)
19. Wang, K., Zhou, S., He, Y.: Growing decision trees on support-less association rules. In: ACM International Conference on Knowledge Discovery and Data Mining, SIGKDD, pp. 265–269. ACM (2000)
20. Wang, W., Yang, J., Yu, P.S.: Efficient mining of weighted association rules (WAR). In: ACM International Conference on Knowledge Discovery and Data Mining, SIGKDD, pp. 270–274. ACM (2000)
21. Xiong, H., Tan, P.N., Koumar, V.: Mining strong affinity association patterns in data sets with skewed support distribution. In: IEEE ICDM, pp. 387–394. ACM (2003)
22. Xiong, H., Tan, P.N., Koumar, V.: Hyperclique pattern discovery. Data Min. Knowl. Discov **13**(2), 219–242 (2006)
23. Younes, N.B., Hamrouni, T., Ben Yahia, S.: Bridging conjunctive and disjunctive search spaces for mining a new concise and exact representation of correlated patterns. In: Pfahringer, B., Holmes, G., Hoffmann, A. (eds.) DS 2010. LNCS, vol. 6332, pp. 189–204. Springer, Heidelberg (2010)
24. Yun, H., Ha, D., Hwang, B., Ho Ryu, K.: Mining association rules on significant rare data using relative support. J. Syst. Softw. **67**(3), 181–191 (2003)