Discovering Useful Patterns from Multiple Instance Data

Article *in* Information Sciences · April 2016

DOI: 10.1016/j.ins.2016.04.007

4 authors:

José María Luna
University of Cordoba (Spain)
40 PUBLICATIONS **327** CITATIONS
SEE PROFILE

Alberto Cano
Virginia Commonwealth University
51 PUBLICATIONS **169** CITATIONS
SEE PROFILE

Virgilijus Sakalauskas
Vilnius University
61 PUBLICATIONS **98** CITATIONS
SEE PROFILE

Sebastian Ventura
University of Cordoba (Spain)
248 PUBLICATIONS **4,764** CITATIONS
SEE PROFILE

Discovering Useful Patterns from Multiple Instance Data

J. M. Luna[a], A. Cano[b], V. Sakalauskas[c], S. Ventura[a,d,*]

[a] *Dpt. of Computer Science and Numerical Analysis, University of Cordoba, Spain*
[b] *Dpt. of Computer Science, Virginia Commonwealth University, Richmond, USA*
[c] *Dpt. of Informatics. Kaunas Faculty of Humanities, Vilnius University, Lithuania*
[d] *Dpt. of Information Systems, King Abdulaziz University, Jeddah, Saudi Arabia*

Abstract

Association rule mining is one of the most common data mining techniques used to identify and describe interesting relationships between patterns from large datasets, the frequency of an association being defined as the number of transactions that it satisfies. In situations where each transaction includes an undetermined number of instances (customers shopping habits where each transaction represents a different customer having a varied number of instances), the problem cannot be described as a traditional association rule mining problem. The aim of this work is to discover robust and useful patterns from multiple instance datasets, that is, datasets where each transaction may include an undetermined number of instances. We propose a new problem formulation in the data mining framework: multiple-instance association rule mining. The problem definition, an algorithm to tackle the problem, the application fields, and the relations' quality measures are formally described. Experimental results reveal the scalability of the problem on different data dimensionality. Finally, we apply it to two real-world applications field: (1) analysis of financial data gathered from one of the most important banks in Lithuania; (2) study of existing relations between records of unemployed gathered from the Spanish public employment service.

Keywords: Association Rules, Multiple-Instance Data, Data Mining

[*] Corresponding author

Email addresses: jmluna@uco.es (J. M. Luna), acano@vcu.edu (A. Cano), virgilijus.sakalauskas@khf.vu.lt (V. Sakalauskas), sventura@uco.es (S. Ventura)

1. Introduction

We are being involved in the digital revolution, which provides the storage of more information on every single topic than ever before. This unrestrained growth of data has led to a situation where the extraction, description and analysis of hidden knowledge is a requirement to be able to take advantage of this massively data storage. Association rule mining (ARM) [1], an unsupervised learning task, is one of the most common data mining techniques [8, 9] used to describe hidden knowledge in data [13]. ARM was proposed as a way for seeking frequent, interesting and strong relations between products in market basket analysis. In this regard, it aims to extract useful knowledge about customers shopping habits to make right decisions.

To date, the market basket analysis has been carried out by considering the whole set of transactions, extracting useful information about relations between products like one of the most well known relationships in this field, i.e., if diaper then beer. Nevertheless, it could easily happen that a specific product could be bought only by a single customer, or even that this single customer usually buys more often than everybody else. Thus, the relations obtained describe a general scenario that provides information that is biased against occasional customers. This issue plays an important role in decision making since once the specific customer stops purchasing, the product will no longer be sold.

In bank financial analysis, a similar analysis can be inferred, where each customer might address a different and varied number of banking transactions. In this field, a traditional ARM analysis provides relations between banking transactions from a general scenario without distinction between frequent and sporadic customers, or even without considering the time stamp in which each transaction was carried out. The distinction between customers, time stamps or any other specific attribute may provide an interesting and quite different analysis with regard to the one inferred from traditional ARM.

This way of describing relations on different scenarios is somehow related to multi-relational learning (MRL) [16] framework, where the scenarios to be described are defined by primary keys. MRL refers to a context where the examples may have a complex internal structure (multiple components and there may be relationships between these components) and this structure is fixed beforehand (primary keys cannot be modified). Nevertheless, the only way of introducing ambiguity to MRL is by means of its primary key, for example, a customer defined by a unique attribute (primary key) can include many alternative instances. The ambiguity problem was introduced by the multiple instance learning (MIL) [6]. In MIL, training transactions are ambiguous and a single transaction may have many alternative instances that describe it. The key challenge in MIL is to cope with the ambiguity, receiving a growing attention in the machine learning community [5, 19].

The goal of this work is to introduce the ambiguity problem into the ARM field in order to solve the problem of discovering relationships which may be pointless in analysis like the previously described. Notice that the ambiguity can be considered from an undefined number of perspectives, grouping the instances by customers, time stamps, or any other different feature, depending on the desired analysis. The proposed solution gives rise to the formulation of a new task, association rule mining on multiple-instance data (MI-ARM), which presents some similar points to MIL. This new task is able to extract useful patterns, and relationships between that patterns, from datasets where each transaction groups a set of instances. To demonstrate the strength and usefulness of using the proposed new task, a sample dataset for the market basket analysis has been considered. Different perspectives have been considered, grouping its records to form multiple instances according to different goals. The knowledge extracted in this example does not describe a real scenario but denotes the high utility of this type of association rules, discovering knowledge that is hardly obtained with traditional ARM algorithms.

In this paper, a series of experiments have been carried out to justify and demonstrate the usefulness of the proposed task, which differs from MRL [25] in its ability to discern groups [26] by any specific feature instead of using a unique attribute. Additionally, our proposed task enables to form group by abstract features, which is impossible in MRL. It is not the intention to compare the results for specific data but to provide an overview on the usefulness of this new problem formulation. In this experimental analysis we also study the scalability of this new problem considering a different number of both transactions and attributes. Finally, two real-world application fields are considered, making an analysis of interesting relations between patterns [4] discovered from real-data.

This paper is arranged as follows. Section 2 presents the most relevant definitions, related work and explains the novelty and contributions of our work correspondingly. Section 3 formally describes the proposed new task. Section 4 describes the proposed algorithm to be considered in this new problem. Section 5 presents the datasets used in the experiments and the results obtained. Finally, some concluding remarks are outlined in Section 6.

2. Preliminaries

In this section, we formally define both the association rule mining (ARM) and multiple-instance learning (MIL) problems and then, we discuss the intuition behind and justification for the new concept and highlight the novelty of our work with regard to some related works.

2.1. Association Rule Mining

The extraction of association rules is considered as an important descriptive task in the data mining field, and it has received enormous attention since its introduction by *Agrawal et al.* [1] in the early 90s. Association rules can be considered as a way to describe interesting and usually hidden associations between items in data [22].

Definition 1 (Association rule). Let $I = \{i_1, i_2, ..., i_n\}$ be the set of items in a dataset, and a pattern P be formally defined as a subset of I, i.e. $\{P = \{i_j, ..., i_k\} \subseteq I, 1 \leq j, k \leq n\}$, that describes valuable features of data. An association rule, defined for a pattern P, is defined as implications of the form **IF** *Antecedent* **THEN** *Consequent*, *Antecedent* (A) and *Consequent* (C) being subsets of the pattern P, i.e. $\{A \subset P \subseteq I \wedge C = P \setminus A\}$, or also $\{C \subset P \subseteq I \wedge A = P \setminus C\}$.

The meaning of an association rule is that if the antecedent is satisfied, then it is highly probable that the consequent is also satisfied. Association rules were originally designed for the market basket analysis as a way to describe relationships between products like $diapers \rightarrow beer$, denoting the high probability of someone buying *diapers* also buying *beer*. It would allow shop-keepers to exploit this relationship by locating the products in an accurate place. Nowadays, more and more application domains [17, 11, 20] use association rules to describe and analyse unknown relations in data, which has given rise to the task of mining association rules.

Definition 2 (Association rule mining). Let $\mathcal{T} = \{t_1, t_2, ..., t_m\}$ be the set of transactions in a dataset, and each transaction $t_i \in \mathcal{T}$ be a subset of items such that $t_j \subseteq I$. The association rule mining (ARM) task aims to find, from the whole set of rules $\mathcal{R}$, any association rule R that satisfies specific user thresholds α of interest, i.e., $ARM = \{\forall\ R \in \mathcal{R}\ :\ quality(R) \geq \alpha\}$.

ARM suffers from different main problems that have been studied by many researchers [7]. First, the discovery of patterns of interest in large datasets can be computationally expensive [10], requiring a large amount of memory [1, 18]. Second, in many application domains, some patterns simply appear by chance [12] and they might be useless for the user. Third, the mining of associations in continuous domains [14] is a hard task and many algorithms, specially exhaustive search algorithms, require a preprocessing step to discretize data.

2.2. Multiple-Instance Learning

The problem of multiple instance learning (MIL) was firstly introduced by *Dietterich et al.* [6] in the context of drug activity prediction. This problem arises in tasks where the training transactions are ambiguous, that is, a single transaction may include many alternative instances that describe it.

Definition 3 (Multiple-instance learning). Let us assume that $\mathcal{T}$ be the set of transactions in data, and each transaction $t_i \in \mathcal{T}$ be composed of a varied number k of instances, i.e. $t_{i,1}$, $t_{i,2}$, ..., $t_{i,k}$. Each of these instances is represented by a distinct feature vector $V(t_{i,j})$, $j \leq k$, so a complete transaction t_i is therefore represented as $\{V(t_{i,1}), ..., V(t_{i,k})\}$. The multiple-instance learning is defined as the looking for a good approximation $\hat{f}$ to f by analysing a set of transactions represented as $\{V(t_{i,1}), V(t_{i,2}), ..., V(t_{i,k})\}$.

According to *Dietterich et al.* [6], the result of f is positive for a transaction t_i if at least one of its instances $t_{i,k}$ has produced a positive result, otherwise, the result of f is negative. *Zucker et al.* [24] determined that to classify a new transaction in the multiple-instance problem by means of a decision tree, the set of instances of the transaction is passed through the tree. As soon as one positive leaf is reached by one of the instances, the transaction is classified as positive, negative otherwise. For other authors, instances are not necessarily alternative descriptions of a transaction but descriptions of different parts of the transaction. Thus, two decision trees (positive and negative) have to be learned. To classify a new transaction, its set of instances is passed through the two trees.

2.3. Contribution and Related Work

The ultimate goal of ARM is the looking for frequent, interesting and strong relations between patterns in a database. In this field, a specific pattern P satisfies a transaction t_j if and only if $P \subseteq t_j$, and the frequency of this pattern $f(P)$ is defined as the number of different transactions that it satisfies, i.e. $f(P) = |\{\forall t_j \in T : P \subseteq t_j\}|$. A pattern P is defined as frequent if and only if the number of transactions that it satisfies is greater or equal to a minimum predefined threshold f_{min}, i.e. $f(P) \geq f_{min}$.

With the increasing interest in the data storage, the extraction of valuable and powerful knowledge is essential. Let us consider a supermarket chain that wants to extract useful knowledge about their customers' shopping habits to make right decisions. It is interesting to obtain this knowledge by analysing each customer regardless the number of transactions that he represents. Thus, data is structured into transactions and each of these transactions (one per customer) comprises an undetermined number of instances (at least one per transaction). A specific pattern can be considered of high interest depending on the number of customers satisfied, so the problem cannot be considered as in classic ARM.

As mentioned in the previous section, this way of storing information is related to multiple instance learning (MIL) [6]. Since many real-world problems can be represented as multi-instance problems, MIL has received a great attention in the machine learning community. MIL has been applied successfully to different domains such as text categorization [2], drug activity prediction [15], web index page recommendation [23] and predicting student performance [21].

The main contribution of this paper is the use of rules to describe behaviours in multiple instance data. We formulate the multiple-instance association rule mining (MI-ARM) problem, which presents some similar points to MIL. This new task can be considered, to some extent, as a simplification of the multi-relational learning (MRL) [16] problem where only a single perspective is considered, that is, only one relation (transactions grouped by a unique attribute) from the multi-relational database. Nevertheless, the comparison between MI-ARM and MRL is pointless since the former includes a set of transactions comprising an undetermined number of instances, and these transactions can be obtained by grouping instances by any feature on a single scenario. On the contrary, MRL refers to a context where the transactions may have a complex internal structure, which is fixed beforehand. Besides, MRL does not allow to group transactions by abstract features as in MI-ARM.

3. Problem Formulation: Multiple-Instance Association Rules

A transactional database comprises a set $\mathcal{T} = \{t_1, t_2, ..., t_n\}$ of transactions where each transaction t_i denotes a specific record in the database. Traditional algorithms for mining association rules [1] work on the transactional database, discovering previously unknown patterns of high interest for a predefined aim. Nevertheless, the increasing interest in data storage has given rise to the extraction of valuable and powerful knowledge. Sometimes, a transactional database can be considered from different perspectives, where each specific transaction provides much richer information since it comprises multiple records or instances, that is, each transaction $t_i \in \mathcal{T}$ comprising a varied number k of instances, i.e. $t_{i,1}$, $t_{i,2}$, ..., $t_{i,k}$. Figure 1 abstractly illustrates how a transactional database that comprises a set of single transactions can be transformed into a varied set of multiple instance databases. Each of these multiple instance databases can be considered as a perspective, since each of its records can be grouped to for a multiple instance.

Traditional association rule mining cannot be applied to extract strong relationships from multiple-instance databases. Hence, this task is only useful for discovering rules in a single perspective, that is, the general scenario in

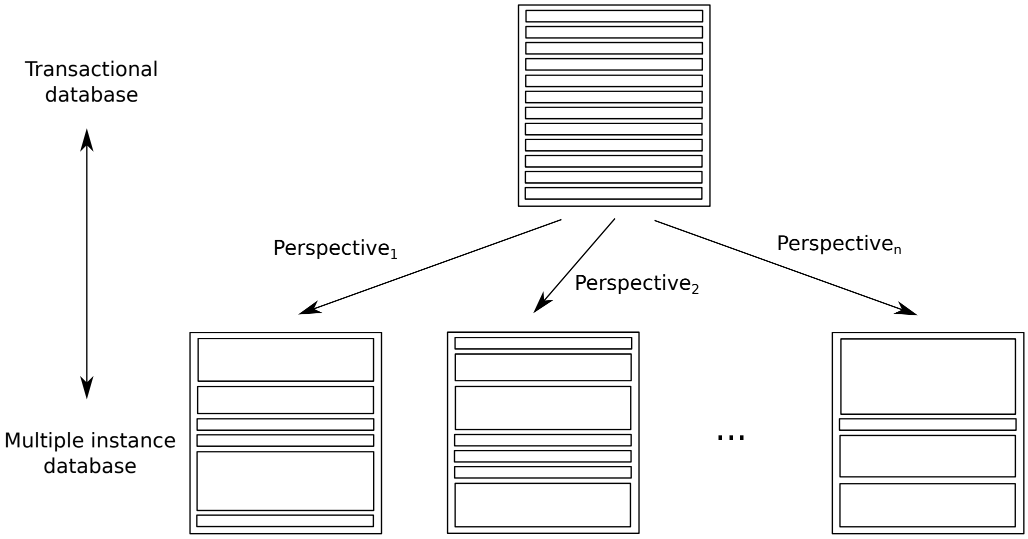


Figure 1: Multiple-instance databases can be obtained from a transactional database by considering different perspectives

which each transaction represents a single record. However, in many application domains, the ability to represents the data into different perspectives is essential, since the knowledge extracted is much richer and of higher interest for the user.

Based on the both previously introduced concepts, we define a new problem that is formulated halfway between ARM and MIL. This new concept, known as multiple-instance association rule mining (MI-ARM), defines the discovery of interesting and strong relations between patterns from multiple-instance data. MI-ARM is presented as an unsupervised learning task where items are analysed from the perspective of transactions including an undetermined number of instances.

Definition 4 (Multiple-instance association rule). Let $\mathcal{T}$ be the set of transactions in data, and each transaction $t_i \in \mathcal{T}$ comprising a varied number k of instances, i.e. $t_{i,1}$, $t_{i,2}$, ..., $t_{i,k}$. Additionally, each single instance $t_{i,j}$ be a subset of items such that $t_{i,j} \in t_i : j \leq k, t_{i,j} \subseteq I$, from the total set of items $I = \{i_1, i_2, ..., i_n\}$ in the dataset. A multiple-instance association rule (MI-AR) is an implication of the type $A \rightarrow C$, A and C being defined from a pattern $P = \{A \cup C\}$, i.e. $\{A \subset P \subseteq I \wedge C = P \setminus A\}$, or also $\{C \subset P \subseteq I \wedge A = P \setminus C\}$.

According to the previous definition (see Definition 4), the frequency of occurrence of a MI-AR obtained from a pattern P is defined as the number of transactions satisfied by P. A transaction t_i is satisfied if and only if at least one instance $t_{i,j}$ is satisfied by P.

Definition 5 (Multiple-instance association rule mining). This task is defined as an unsupervised learning problem. The aim of mining multiple-instance association rules (MI-ARM) is to find, from a multiple instance dataset, any MI-AR R that satisfies specific user thresholds α of interest, i.e.MI-ARM $= \{\forall\ R\ :\ \mathit{quality}(R) \geq \alpha\}$.

MI-ARM (see Definition 5) considers a set of quality measures to determine the interest of the rules mined. The main difference between ARM and MI-ARM falls upon the concept of transactions and instances. The support quality measure (Equation 1) of a rule $A \rightarrow C$ obtained from a pattern P, i.e. $P = \{A \cup C\}$, is defined as the number of transactions satisfied by P. A transaction t_i is satisfied if and only if at least one instance $t_{i,j}$ is satisfied by P.

$$Support_{MI-ARM}(A \cup C) = \frac{|\{\forall t_i \in \mathcal{T} \ : \ \exists t_{i,j} \in t_i \wedge P = \{A \cup C\} \subseteq t_{i,j}\}|}{|\mathcal{T}|} \tag{1}$$

A major feature of the support quality measure in the field of MI-ARM is that it is possible to obtain A and C that both satisfy all the transactions by isolation, but if they are analysed together they do not satisfy any transaction, i.e. $A \cap C = \emptyset$. On the contrary, this assertion is not possible in classic ARM since it considers each transaction as a single instance.

As for the confidence measure (Equation 2), it is defined as the proportion of the number of transactions that include A and C among all the transactions that comprise A, which is determined by the support of A.

$$Confidence_{MI-ARM}(A \cup C) = \frac{Support_{MI-ARM}(A \cup C)}{Support_{MI-ARM}(A)} \tag{2}$$

Support and confidence are broadly conceived as the finest quality measures in ARM and, consequently, a great variety of proposals make use of them by using certain minimum thresholds. Nevertheless, many authors have considered that the mere fact of exceeding these quality thresholds does not guarantee that the rules are interesting at all [3], and some different quality measures have been proposed.

The lift quality measure (see Equation 3) establishes how many times A and C occur together more often than would be expected if they were statistically independent.

$$Lift_{MI-ARM}(A \cup C) = \frac{Support_{MI-ARM}(A \cup C)}{Support_{MI-ARM}(A) \times Support_{MI-ARM}(C)} \tag{3}$$

The conviction metric (see Equation 4) represents the ratio of the expected frequency that A occurs without C, considering A and C statistically independent sets, divided by the observed frequency of incorrect predictions.

$$Conviction_{MI-ARM}(Y) = \frac{1 - Lift_{MI-ARM}(A \cup C)}{1 - Confidence_{MI-ARM}(A \cup C)} \tag{4}$$

In a similar way to the lift measure, Leverage (see Equation 5) calculates the proportion of additional cases covered by both A and C above those expected if A and C were independent of each other.

$$
\begin{aligned}
Leverage_{MI-ARM}(A \cup C) = Support_{MI-ARM}(A \cup C) - \\
(Support_{MI-ARM}(A) \times Support_{MI-ARM}(C))
\end{aligned} \tag{5}
$$

Let us consider a dataset (see Table 1) for the market basket analysis, which comprises information about products purchased by customers in a supermarket. This sample dataset includes information about twelve different transactions comprising a set of different products. As previously described, the same dataset can be analysed on multiple perspectives and MI-ARs can be applied to any of these scenarios. This issue implies the strength of this new task, since the market basket analysis can be carried out by multiple perspectives depending on the goal of the managers.

Imagine that the managers require to carry out a market basket analysis to obtain information about how the products are related for the sake of improving the sales. In this regard, a descriptive analysis is performed by on the general scenario, discovering relations such as IF Teenager THEN Tuna (Support=0.5, Confidence=1.0). This association rule states that every Teenager purchases Tuna each time that he/she comes to this supermarket. A manager could advertise Tuna between Teenagers to increase profits.

There are many other descriptive analyses that could be done with the same dataset and none of the existing ARM algorithms could be carried out. For instance, let us now consider that the managers want to analyse the relationships by group age (see Table 2), in such a way that relations between products are obtained regardless the age of people that usually buy in the supermarket. Considering data in this new perspective new associations

Table 1: Sample dataset for the market basket analysis

Transaction	**Time stamp**	**ID**	**Age group**	**Fresh fruit**	**Seafood**
1	April	1	Senior	Orange	Tuna
2	April	2	Teenager	Banana	Tuna
3	April	4	Teenager	Banana	Tuna
4	June	1	Senior	Orange	Anchovy
5	June	3	Adult	Banana	Tuna
6	July	1	Senior	Orange	Anchovy
7	July	2	Teenager	Banana	Tuna
8	July	2	Teenager	Orange	Tuna
9	July	4	Teenager	Banana	Tuna
10	December	2	Teenager	Banana	Tuna
11	December	3	Adult	Orange	Anchovy
12	December	1	Senior	Orange	Anchovy

Table 2: Sample multiple instance dataset for the market basket analysis by using the perspective of **Age group**

Transaction	Time stamp	ID	Age group	Fresh fruit	Seafood
1	April	1	Senior	Orange	Tuna
	June	1	Senior	Orange	Anchovy
	July	1	Senior	Orange	Anchovy
	December	1	Senior	Orange	Anchovy
2	June	3	Adult	Banana	Tuna
	December	3	Adult	Orange	Anchovy
3	April	2	Teenager	Banana	Tuna
	April	4	Teenager	Banana	Tuna
	July	2	Teenager	Banana	Tuna
	July	2	Teenager	Orange	Tuna
	July	4	Teenager	Banana	Tuna
	December	2	Teenager	Banana	Tuna

can be obtained when the group age is considered to cluster instances into transactions. In this regard, the rule IF April THEN Tuna (Support=0.66, Confidence=1.0) determines that April and Tuna are features that usually appears together regardless the group age of the customer. Similarly, the rule IF July THEN Orange denotes that July is a good month to sell Orange.

Continuing with the previous sample dataset, it is also possible to form groups of transactions by using abstract attributes that are not inherently in the dataset. For instance, let us consider that managers want to obtain information about existing relationships between products regardless the season (Spring, Summer or Winter), so the new multiple-instance database is organized as shown in Table 3. Considering this perspective, it is possible to

Table 3: Sample multiple instance dataset for the market basket analysis formed by abstract attributes that are not inherently in the dataset, for example, the season.

Transaction	Time stamp	ID	Age group	Fresh fruit	Seafood
1	April	1	Senior	Orange	Tuna
	April	2	Teenager	Banana	Tuna
	April	4	Teenager	Banana	Tuna
2	June	1	Senior	Orange	Anchovy
	June	3	Adult	Banana	Tuna
	July	1	Senior	Orange	Anchovy
	July	2	Teenager	Banana	Tuna
	July	2	Teenager	Orange	Tuna
	July	4	Teenager	Banana	Tuna
3	December	1	Senior	Orange	Anchovy
	December	3	Adult	Orange	Anchovy
	December	2	Teenager	Banana	Tuna

discover the rule IF Tuna THEN Teenager (Support=1.0, Confidence=1.0), which determines that in all the seasons exist at least one customer who buys tuna and he/she is a teenager. From this knowledge, it is stated that tuna is not a seasonal product as, for example, anchovy is. Notice that anchovy is not usually sold in Spring, presenting a support value of 0.66, that is, it is sold in 2 out 3 seasons.

All of this demonstrate the usefulness of using multiple-instance in association rule mining, discovering knowledge that is hardly obtained with traditional ARM algorithms. Finally, it should be noted that we have considered a sample dataset to understand the strength of using MI-ARM. In this regard, the knowledge extracted in this example does not describe a real scenario but denotes the high utility of this type of association rules.

4. Algorithm: Apriori-MI

In this work, we have introduced a new problem that is formulated halfway between ARM and MIL so that the goal is to deal with multiple-instance datasets from an unsupervised and descriptive point of view. To date, there is no definition in this field, which could be considered as an unexplored area. In this regard, we consider that this new problem requires to be solved and analysed by considering key algorithms like Apriori, setting the basis for further algorithms and leaving it open to further research studies.

In the ARM field, the best-known algorithm was proposed by *Agrawal* [1] at the beginning of the 90s. This algorithm, known as Apriori, is based on an exhaustive search methodology and divides the ARM problem into two subproblems: (a) obtaining all the frequent patterns in data and (b) extracting all the ARs — according to a predefined minimum confidence threshold β — starting from the results obtained in the previous step. While the second step is straight forward, the first step needs more attention since finding all the frequent patterns in data is not a trivial task. It involves the searching for any feasible pattern, so considering a dataset containing n single-items, the set of available patterns is $2^n - 1$ (excluding the empty set which is not a valid pattern). Although the size of the set grows exponentially in the number of patterns, an efficient search can be considered by using the downward-closure property of support (also known as anti-monotone). This property determines that if a length-k pattern is not frequent in data, none of its length-$(k + 1)$ super patterns can be frequent.

For a better understanding, let us consider a dataset comprising the transactions {a,c,d}, {b,c,d}, {b,d}, {c,d}, {b,c,d}. The first step considered by the Apriori algorithm is to calculate the number of transactions where each single-item is satisfied, that is, the absolute support or frequency of occurrence of each item separately ($\{a\} = 1$, $\{b\} = 3$, $\{c\} = 4$ and $\{d\} = 5$). In the next steps, Apriori generates a list of all 2-pairs, 3-pairs, and so on. As depicted in Figure 2, the process of mining patterns can be very time-consuming, so the downward-closure property plays an important role by pruning branches of the tree. Let us assume that the minimum support α to qualify a pattern as frequent is 2, which depends on the context. Thus, any super-set that comprises the single-item $\{a\}$, which is satisfied by only one transaction, will be infrequent and can be pruned.

This methodology cannot be directly applied to the multiple-instance problem since it deals with a set of transactions $\mathcal{T}$ and each transaction $t_i \in \mathcal{T}$ having a varied number k of instances, i.e. $t_{i,1}$, $t_{i,2}$, ..., $t_{i,k}$. Definitions previously described for the MI-ARM concept determines that a transaction t_i is satisfied if and only if at least one instance $t_{i,j} \in t_i$ is satisfied by $P = \{A \cup C\}$, which form the rule $A \to C$. In this sense, the original Apriori algorithm is modified to deal with multiple instance datasets, giving

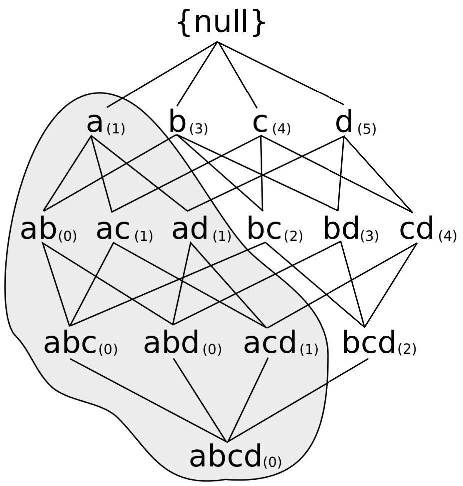


Figure 2: Steps of the Apriori algorithm for mining frequent patterns. Shaded area represents infrequent patterns that can be removed according to the downward-closure property considering 2 as the minimum value of frequency

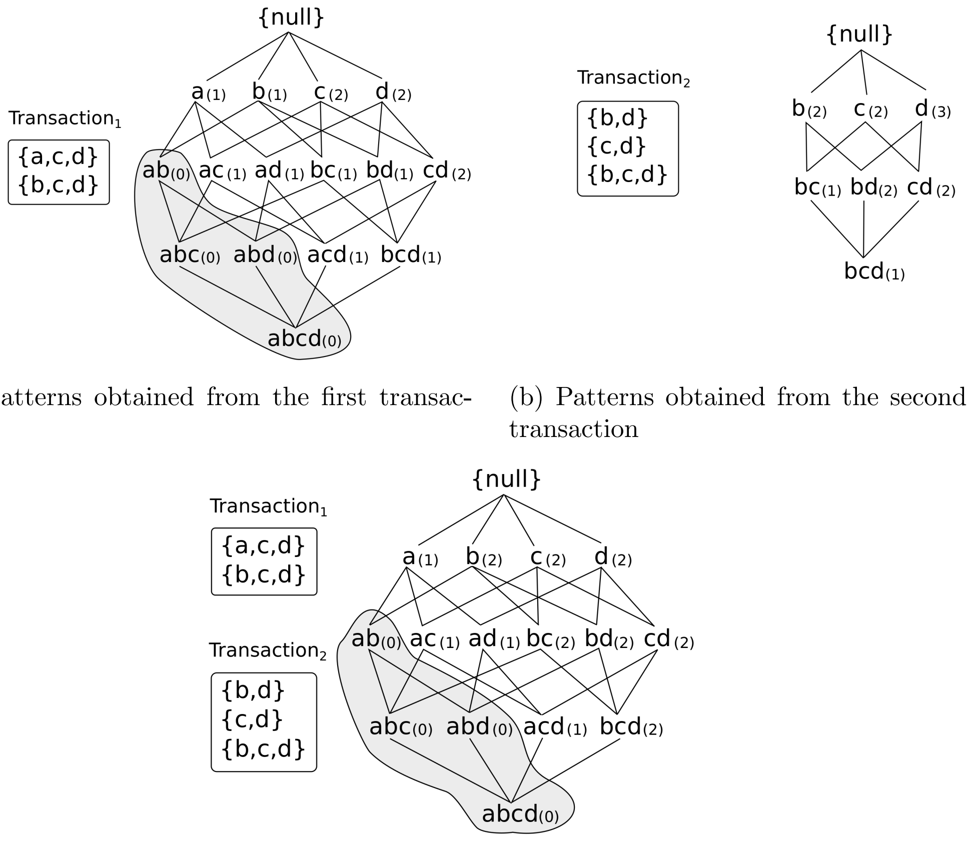
rise to the Apriori-MI algorithm. Focusing on the Apriori procedure (see Algorithm 1), it requires to iterate every transaction in the dataset and each of these transactions includes a different set of instances that also need to

Algorithm 1 Pseudo-code of the Apriori-MI algorithm

```
Input: $\mathcal{E}, \alpha$
Output: $\mathcal{L}$
    procedure Apriori algorithm for multiple-instance problems
    $k = 1$
    $\mathcal{L}_1 \leftarrow \{k - patterns\}$
    $\mathcal{C} \leftarrow \emptyset$
    $\mathcal{A} \leftarrow \emptyset$
    $k = 2$
    while $\mathcal{L}_{k-1} \neq \emptyset$ do
        $\mathcal{C}_k \leftarrow \{a \cup \{b\} | a \in \mathcal{L}_{k-1} \wedge b \in \bigcup \mathcal{L}_{k-1} \wedge b \notin a\}$
        for all transaction $t_i \in \mathcal{T}$ do
            for all instance $t_{i,j} \in t_i$ do
                if $\exists\ c | c \in \mathcal{C}_k \wedge c \subseteq t_{i,j}$ then
                    $\mathcal{A}_e \leftarrow c$
                    $count[c] \leftarrow count[c] + 1$
                    break
                end if
            end for
        end for
        $\mathcal{L}_k \leftarrow \{c | c \in \mathcal{C}_k \wedge count[c] \geq \alpha\}$
        $k \leftarrow k + 1$
    end while
    $\mathcal{L} \leftarrow \bigcup_{k} \mathcal{L}_k$
    return $\mathcal{L}$
    end procedure
Input: $\mathcal{L}, \beta$
Output: $\mathcal{R}$
    procedure Rule generation procedure
    $\mathcal{R} = \emptyset$
    for all $l \in \mathcal{L}$ do
        for all $x \subset l$ such that $x \neq \emptyset$ and $x \neq l$ do
            if $support(l)/suppor(x) \geq \beta$ then
                $\mathcal{R} \leftarrow \mathcal{R} \cup \{x \rightarrow (l - x)\}$
            end if
        end for
    end for
    return $\mathcal{R}$
    end procedure
```

be analysed to determine its coverage (see lines 8 to 16, Algorithm 1). A really important feature of this evaluation process is that Apriori-MI does not require to check all the instances within each transaction. Once a specific instance is satisfied, then the remaining ones are not analysed (see lines 10 to 14, Algorithm 1) and the algorithm continues with the next transaction.

For a better understanding, consider the transactions {{a,c,d}, {b,c,d}} and {{b,d}, {c,d}, {b,c,d}}, and a threshold value of 1, the Apriori-MI mines patterns as shown in Figure 3. In this algorithm the downward-closure prop-



(a) Patterns obtained from the first transaction

(b) Patterns obtained from the second transaction

(c) Patterns obtained from the resulting patterns of the set of transactions

Figure 3: Apriori-MI algorithm by using a minimum threshold value of one. Shaded area represents infrequent patterns that can be removed according to the downward-closure property

erty can be applied to discard misleading patterns and to reduce the computational time. For each transaction, Apriori-MI carries out an exhaustive search (Figures 3(a) and 3(b)), obtaining any pattern that satisfies the minimum threshold value. The resulting set of patterns is obtained from the sum of the sets previously obtained for each transaction (see Figure 3(c)), considering as frequent those that satisfy most transactions. It should be noted that each specific pattern is computed only once for each transaction.

In a second step and similarly to Apriori, the proposed Apriori-MI algorithm discovers any MI-AR from the set of frequent patterns. Thus, the set of patterns *ac*, *ad*, *bc*, *bd*, *cd*, *acd* and *bcd* are used to obtain a resulting set of rules. This set of rules comprises all the possible combinations from the set of frequent patterns: $a \rightarrow c$, $c \rightarrow a$, $a \rightarrow d$, $d \rightarrow a$, $b \rightarrow c$, $c \rightarrow b$, $d \rightarrow b$, $b \rightarrow d$, $d \rightarrow c$, $c \rightarrow d$, $a \rightarrow c \wedge d$, $c \rightarrow a \wedge d$, $d \rightarrow a \wedge c$, $c \wedge d \rightarrow a$, $a \wedge d \rightarrow c$, $a \wedge c \rightarrow d$, $b \rightarrow c \wedge d$, $c \rightarrow b \wedge d$, $d \rightarrow b \wedge c$, $c \wedge d \rightarrow b$, $b \wedge d \rightarrow c$, and $b \wedge c \rightarrow d$.

5. Experimental Analysis

In this section, we demonstrate the importance of MI-ARM by applying it to a series of datasets. It is not our intention to compare results of different algorithms but to provide an overview of the use of multiple-instance as an unsupervised and descriptive task. First, we study the utility of using MI-ARM by comparing results with regard to classic ARM. Then, we analyse the computational cost by considering a set of artificial data comprising different number of transactions, instances per transaction and attributes.

5.1. Strength of using MI-ARM

In this section, we carry out an in depth description about the strength of using the Apriori-MI algorithm by considering the same sample market basket dataset used along this paper (see Table 1). Results are compared to the traditional Apriori algorithm, which was proposed to be applied on transactional datasets.

Let us consider now that the managers require to carry out a market basket analysis to obtain information about how the products are related for the sake of improving the sales. In this regard, a descriptive analysis is performed by using either the traditional Apriori algorithm and the proposed Apriori-MI algorithm for mining frequent relationships on the general scenario (see Table 1). A minimum support threshold of 50% (6 transactions) and a minimum confidence threshold of 70% are considered. Once the Apriori algorithm is executed, two frequent patterns were discovered: {Teenager, Tuna} and {Banana, Tuna}. From these two frequent patterns it is possible to obtain four different association rules to describe the data behaviour:

Rule	Support	Confidence
IF Teenager THEN Tuna	50%	100%
IF Tuna THEN Teenager	50%	75%
IF Tuna THEN Banana	50%	75%
IF Banana THEN Tuna	50%	100%

Considering the most accurate association rules discovered, it is possible to state that every teenager purchases tuna each time that he/she comes to this supermarket. There is no restriction in this rule to be satisfied, so a manager can advertise tuna between teenagers to increase profits. Managers can also consider both bananas and tuna as a pack to be sold since everyone that buys banana also buys tuna. As shown, this is the only information that anybody could obtain considering the support restrictions since no more relationships could be discovered according to the predefined thresholds.

Nevertheless, there are many other descriptive analyses that can be done with the same dataset and none of the existing ARM algorithms could be carried out. For instance, consider that the managers want to analyse the relationships by group age (see Table 2), in such a way that relations between products are obtained by Apriori-MI regardless the age of people that usually buy in the supermarket. Considering data in this new perspective and using the same support threshold, that is, a minimum value of 50%, the following patterns of size one are obtained: {April}, {June}, {July}, {December}, {Banana}, {Orange}, {Anchovy} and {Tuna}. In the same way, the following patterns of size 2 are obtained: {April, Tuna}, {July, Orange}, {December, Orange}, {December, Anchovy}, {Banana, Tuna}, {Orange, Tuna} and {Orange, Anchovy}. Finally, the following pattern of size 3 is obtained: {December, Orange, Anchovy}. From this set of frequent patterns, the following association rules to describe the data behaviour are obtained:

Rule	Support	Confidence
IF April THEN Tuna	66%	100%
IF July THEN Orange	66%	100%
IF December THEN Orange	66%	100%
IF Anchovy THEN December	66%	100%
IF Banana THEN Tuna	66%	100%
IF Anchovy THEN Orange	66%	100%
IF Orange AND Anchovy THEN December	66%	100%
IF Orange AND December THEN Anchovy	66%	100%
IF Anchovy AND December THEN Orange	66%	100%
IF Anchovy THEN Orange AND December	66%	100%

These results are quite interesting to understand the usefulness of the analysis of the data by considering different perspectives. First, the rules obtained are different to the ones obtained by the traditional Apriori algorithm, and the knowledge extracted is also different. For instance, the rule IF Tuna THEN Banana, obtained by Apriori, determined that both banana and tuna were frequent when they appear together. However, this rule is not satisfied when we analyse the dataset by group age and using Apriori-MI, denoting the high correlation between these two products and the group age. Thus, once one of the existing group age stop purchasing the products, the relationship is altered. On the contrary, the rule IF Banana THEN Tuna is satisfied regardless the age group of the customers with an accuracy of 100% in both cases, that is, considering and without considering any group age.

Second, new associations are obtained by Apriori-MI when the group age is considered to cluster instances into transactions. In this regard, the rule IF April THEN Tuna determines that April is a good month to sell Tuna in this supermarket. Similarly, the rule IF July THEN Orange denotes that July is a good month to sell Orange. These two associations were not discovered by the traditional Apriori algorithm since they were satisfied by only 25% and 16% of the transactions, respectively.

Similarly to the previous example, in which transactions were grouped by a specific attribute (age), the multiple instance databases might be obtained by considering other attributes or even by means of an abstract attributes that are not inherently in the dataset. For instance, consider that the managers want to obtain information about existing relationships between products regardless the season (Spring, Summer or Winter), so the new multiple-instance database is organized as shown in Table 3.

Analysing the dataset by considering the predefined abstract attribute (season) and considering the same support threshold, that is, a value of 50%, the following patterns of size 1 are obtained: {ID 1}, {ID 2}, {ID 3}, {ID 4}, {Senior}, {Teenager}, {Adult}, {Orange}, {Banana}, {Tuna} and {Anchovy}. In the same way, the following patterns of size 2 are obtained: {ID 1, Senior}, {ID 2, Teenager}, {ID 3, Adult}, {ID 4, Teenager}, {ID 1, Orange}, {ID 1, Anchovy}, {ID 2, Banana}, {ID 2, Tuna} and {ID 4, Banana}, {ID 4, Tuna}, {Teenager, Banana}, {Teenager, Tuna}, {Senior, Orange}, {Senior, Anchovy}, {Orange, Tuna}, {Orange, Anchovy}, {Banana, Tuna}. Considering the patterns of size 3, the following sets are obtained: {ID 1, Senior, Orange}, {ID 1, Senior, Anchovy}, {ID 2, Teenager, Banana}, {ID 2, Teenager, Tuna}, {ID 4, Teenager, Banana}, {ID 4, Teenager, Tuna}, {Teenager, Banana, Tuna} and {Senior, Orange, Anchovy}. Finally, the following patterns of size 4 are obtained: {ID 1, Senior, Orange, Tuna}, {ID 2, Teenager, Banana, Tuna} and {ID 4, Teenager, Banana, Tuna}.

Considering the aforementioned group of frequent patterns, a set of association rules are obtained. In this regard, we have chosen a subgroup of association rules that could be of interest to be described:

Rule	**Support**	**Confidence**
IF Teenager THEN Tuna and Banana	100%	100%
IF Tuna THEN Teenager	100%	100%
IF Tuna THEN Banana	100%	100%
IF Anchovy THEN Orange	66%	100%

Analysing the aforementioned rules, it is discovered that, whereas traditional Apriori determines that 75% of the customers that buy tuna are teenagers, the fact of considering the season of the year as an abstract attribute to obtain transactions modifies the knowledge extracted. In this regard, Apriori-MI determines that in all the seasons exist at least one customer who buys tuna and he/she is a teenager. Thus, the aforementioned rule calculates an accuracy of 100% instead of 75% as it was obtained by Apriori. From this knowledge, it is stated that tuna is not a seasonal product as, for example, anchovy is. Notice that anchovy is not usually sold in Spring, presenting a support of 66%, that is, it is sold in 2 out 3 seasons.

To sum up, we have demonstrated the usefulness of using multiple-instance in association rule mining, discovering knowledge that is hardly obtained with traditional ARM algorithms.

5.2. Experimental results and computational cost

The aim of this section is to analyse the scalability of the MI-ARM task once that its strength was described in the previous subsection. In this regard, we have considered the use of a varied set of 30 datasets with different features[1]. In this study, we have included datasets that comprise between 200 and 5000 different transactions, and the average number of instances per transaction varies between 3.5 to almost 15. Additionally, we have considered

[1] The datasets and the description of their features (number of transactions, attributes, average number of instances per transaction, etc) are available at `http://www.uco.es/grupos/kdis/kdiswiki/MI-ARM`.

a different number of attributes and values per attribute, considering datasets that comprise between 30 to 240 different values. Despite the fact that there are many multiple-instance datasets in the literature, we have considered this set of artificial datasets to analyse, in a correct way, the performance of the algorithm when the number of transactions, attributes and values per attribute varies. In this regard, the computational cost could be well studied since the features of the data are accordingly predefined for the study.

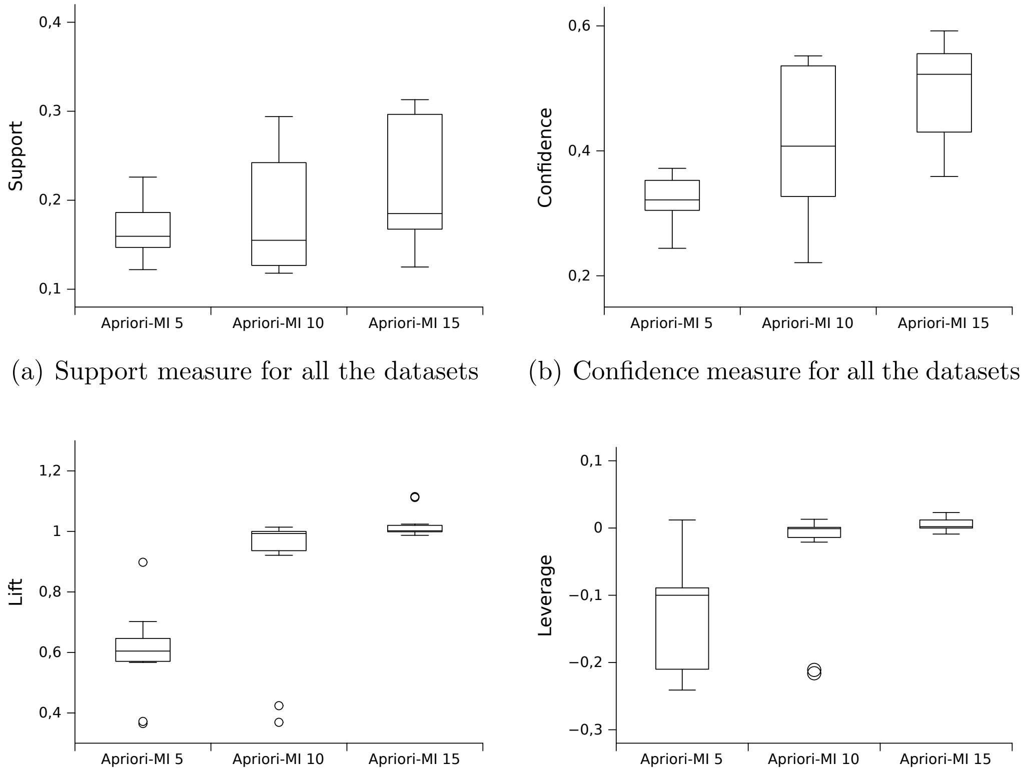
Table 4 shows the results obtained for the set of datasets. For each quality measure, the Apriori-MI algorithm is run using a minimum support threshold value of 0.1, so no rule having a lower support value is considered by the algorithm. As for the confidence value, no threshold is considered.

Analysing the average results depicted in Table 4, the number of rules

Table 4: Number of rules and average values for five quality measures when Apriori-MI is applied to a varied set of data.

Dataset	**# Rules**	**Support**	**Confidence**	**Lift**	**Conviction**	**Leverage**
#1	686	0.155	0.319	0.653	0.760	-0.100
#2	192	0.156	0.408	1.010	1.019	0.013
#3	160	0.125	0.404	1.115	1.098	0.012
#4	1480	0.147	0.324	0.702	0.796	-0.086
#5	262	0.154	0.407	1.014	1.026	0.001
#6	208	0.128	0.406	1.112	1.101	0.012
#7	2978	0.127	0.307	0.626	0.766	0.093
#8	140	0.240	0.548	0.994	1.003	-0.002
#9	210	0.167	0.502	0.994	1.011	-0.001
#10	36788	0.147	0.244	0.365	∞	-0.176
#11	13966	0.129	0.250	0.424	0.498	-0.211
#12	430	0.313	0.560	0.987	0.996	-0.009
#13	7392	0.213	0.359	0.567	∞	-0.227
#14	294	0.294	0.542	0.993	1.016	-0.007
#15	276	0.299	0.586	1.008	1.012	0.004
#16	26670	0.172	0.263	0.372	0.392	-0.241
#17	12964	0.120	0.221	0.369	0.475	-0.216
#18	538	0.289	0.538	0.998	1.002	0.001
#19	19998	0.191	0.334	0.586	0.561	-0.193
#20	35979	0.118	0.342	0.993	1.321	0.000
#21	422	0.176	0.359	1.002	1.224	0.001
#22	15523	0.164	0.366	0.898	0.982	0.012
#23	12033	0.126	0.322	0.921	1.002	0.012
#24	528	0.194	0.542	1.001	1.005	0.023
#25	1916	0.122	0.304	0.623	0.764	-0.092
#26	100	0.243	0.552	1.002	1.017	-0.001
#27	150	0.169	0.507	1.002	1.019	-0.001
#28	4498	0.226	0.372	0.581	∞	-0.232
#29	276	0.277	0.518	0.982	∞	-0.021
#30	224	0.309	0.592	1.024	1.029	0.002

discovered is higher when the number of values per attribute is lower. This is caused by the higher number of transactions that could be satisfied when the attributes comprise a small number of values. On the contrary, the mere fact of considering a higher number of values per attribute implies that the attributes or patterns are hardly satisfied so the number of transactions comprising specific patterns is lower. Additionally, in some datasets, e.g. *dataset #10*, *dataset #13*, *dataset #28* and *dataset #29*, the average value computed for the conviction measure is denoted as infinity. This determines that, in at least one of the discovered rules, the confidence value is maximum, i.e. value 1, so according to the conviction measure (see Equation 4), the denominator of the equation is zero and the result is ∞. Figure 4 includes a



(a) Support measure for all the datasets

(b) Confidence measure for all the datasets

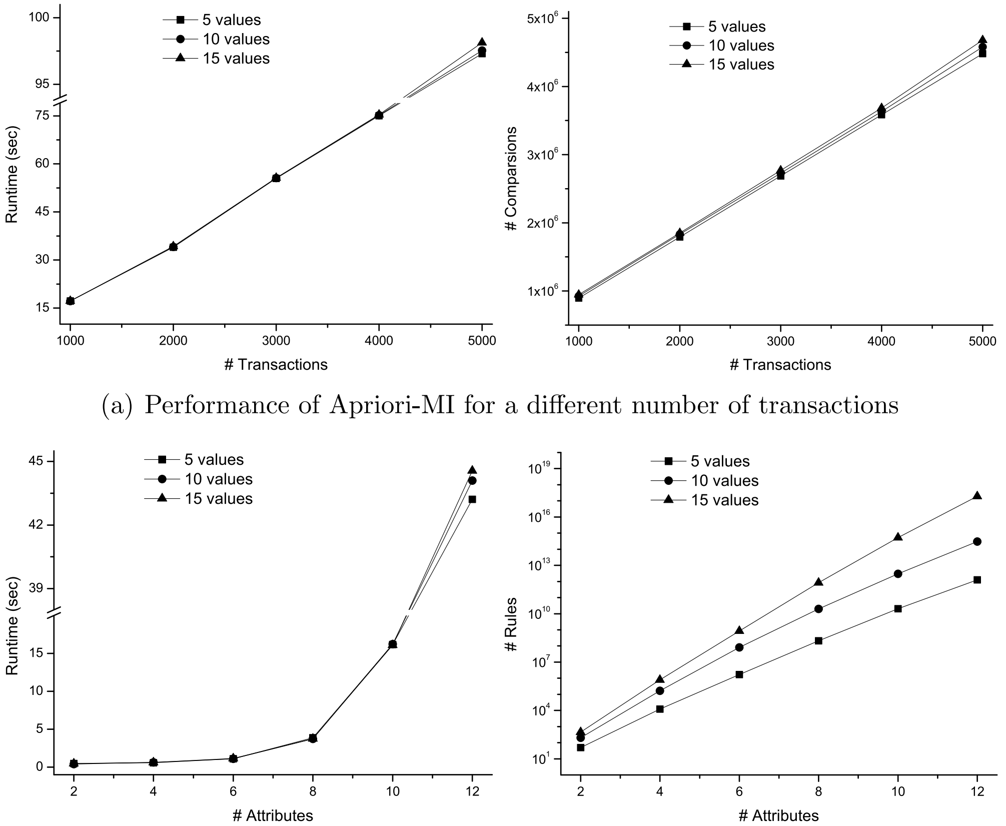
(c) Lift measure for all the datasets

(d) Leverage measure for all the datasets

Figure 4: Boxplot of different quality measures for all the datasets, considering 5, 10 and 15 different values per attribute

set of boxplot graphics that show the average values obtained for the quality measures when a different number of values per attribute (5, 10 and 15) is considered.

Finally, we carry out an analysis of the scalability of the proposed task (see Figure 5). First, it should be noted that the scalability of ARM algorithms have been widely studied by different researchers [14], demonstrating that the runtime of exhaustive search algorithms exponentially increases with the number of attributes. In this study, we aim to analyse the performance of the proposed algorithm for mining association rules in multiple



(a) Performance of Apriori-MI for a different number of transactions

(b) Performance of Apriori-MI for a different number of attributes

Figure 5: Analysis of the scalability of the Apriori-MI algorithm for different number of both transactions and attributes. The analysis is carried out by using datasets that comprise attributes with 5, 10 and 15 values each one.

instance datasets. In this regard, Figure 5 shows the scalability of the proposed exhaustive search MI-ARM algorithm when it is applied to datasets with different number of both transactions (Figure 5(a)) and attributes (Figure 5(b)). As execution time can be affected by numerous parameters and highly depends on the hardware used, we have illustrated the results by using both the runtime in seconds and a different metric —the number of comparisons required to mine the association rules and the amount of rules to be discovered.

Considering the performance of Apriori-MI when the number of transactions increases (Figure 5(a)), a linear growth is obtained, and there is no high differences among datasets with different number of values per attribute. The increasing number of transactions affects the runtime due to an increment in the number of comparisons required to calculate the measures for each rule. Thus, Figure 5(a) also shows how the number of transactions affects the number of comparisons required.

Finally, considering the variation of the number of attributes, Figure 5(b) illustrates that the runtime exponentially increases with the increment of the number of attributes. Additionally, the number of attributes also has a huge influence on the amount of rules to be discovered. Figure 5(b) also shows the number of feasible rules to be mined for a different number of both attributes and values per attribute.

6. Real Cases of Use

Finally, in this section, we apply MI-ARM to two real-world problems, describing the importance of MI-ARM in both fields. We first apply MI-ARM to the financial field, considering data gathered from an important bank in Lithuania. Second, we apply the new proposal in accordance with the Spanish public employment service.

6.1. Application to the financial field

In this section, we apply the new problem formulation to a real-world problem. The aim is to demonstrate the applicability of the proposed problem regardless the algorithm used to this end, so we use a multiple-instance dataset including 13,811 transactions with information about banking operations where each user is defined as a transaction comprising different operations for a specific day.

Data description. Data have been gathered from an important bank in Lithuania, and comprise 13,811 transactions with 61,346 instances in general, so the average number of instances per transaction is 4.44. The dataset is organised in features or attributes including: currency of the operations (Swiss franc, Danish krone, Pound sterling, Latvian Lat, Norwegian krone, Polish zloty, Lithuanian litas, Russian ruble and Euro); amount local currency; product application (personal account, overdraft, special deposits, child, account, mortgage savings account, provisions for loans, etc); account officer id; and sector (bank, large corporate, medium corporate, other financial institution, charitable institutions, private person, etc). The goal is to describe the data to determine interesting and strong relations between operations to detect frequent financial habitats in the population.

Data have been organized into transactions that represent different bank customers. Each transaction (customer) comprises a different number of instances, one per successful banking operation. The data organization enables MI-ARM to be applied on it, providing a description based on the customers' habits in such a way that sporadic customers are as important as habitual customers.

By using MI-ARM, an institution is able to improve their needs and offer specific products to specific clients. Additionally, the description could be used to detect anomalous operations and to avoid banking frauds. The goal of this analysis is just to demonstrate the usefulness of the proposed MI-ARM formulation to a real-world field.

Results. Applying MI-ARM formulation to the aforementioned dataset we obtained a series of accurate rules that relate interesting clients' behaviours. Table 5 shows some MI-ARs obtained, depicting the support, confidence and lift values. It should be noted that a pattern is satisfied in a specific transaction if and only if at least one instance within the transaction is satisfied.

First, we obtain two interesting MI-ARs that are somehow related: (1) IF (amountlcy $>$ -172,034.22 AND currency = LTL) THEN (productCategory = current account), and (2) IF (currency = LTL) THEN (productCategory = current account). The first rule is a specialization of the second one, which determines that 94.16% of the banking operations involving Lithuanian litas (LTL) are related to the current account of the clients. Both rules are satisfied in the 88.34% of the transactions with a confidence of 0.94 and a lift value of 1.025. The first rule determines that those transactions that involve an

Table 5: Multiple-instance association rules obtained from data of a bank of Lithuania. Support and confidence are represented in per unit basis.

Rule	**Support**	**Confidence**	**Lift**

IF (amountlcy = [-172,034.22, max]
AND currency = LTL)
THEN (productCategory = current account)
	0.883	0.942	1.025

IF (currency = LTL)
THEN (productCategory = current account)
	0.883	0.942	1.025

IF (sector = individuals)
THEN (amountlcy = [-31,512,532.83, max])
	0.201	1.000	1.001

IF (currency = LTL)
THEN (amountlcy = [min, 48,255.16])
	0.873	0.999	1.021

amount higher than -172,034.22 LTL are related to the current account of the clients. In other words, it could be said that it is quite difficult to find a Lithuanian litas account whose amount of currency is lower than -172,034.22 LTL.

Second, we discover a very strong MI-AR that relates a specific sector to the amount of currency used in the transaction, i.e. IF (sector = individuals) THEN (amountlcy $>$ -31,512,532.83). This rule, which is satisfied in 20.11% of the transactions, is obtained with a maximum confidence value, denoting that when the client that has ordered the banking operation is an individual (he/she does not act as a company or institution) then the amount of currency is greater than -31,512,532.83. This rule means than, in general, lower amount of currency is always managed by institutions or companies instead of individuals.

Finally, an interesting MI-AR is discovered, IF (currency = LTL) THEN (amountlcy $\leq$ 48,255.16), denoting that the amount of money used in operations that involve LTL as currency is always lower than 48,255, and this rule presents an accuracy of 99.92%. This rule could give rise to determine that a specific operation involving more than 48,255 Lithuanian litas is considered outrageous and should be carefully revised since it could be a fraud.

In this analysis, it is also possible to obtain negative association rules, which denotes a negative relationship between the antecedent and consequent of the rule. For instance, the three following sample rules are negative associations: (1) IF(currency $\neq$ LVL) THEN (productCategory $\neq$ charge accumulation); (2) IF (currency $\neq$ SEK) THEN (productCategory $\neq$ charge accumulation); (3) IF (currency $\neq$ CHF) THEN (productCategory $\neq$ charge accumulation). All of these rule presents a confidence value of 1.00, denoting that none of the operations that use Latvian lats (LVL) are related to a charge accumulation. In the same way, none of the operations that use Swedish krona (SEK) or Swiss franc (CHF) are related to a charge accumulation. The support of the rules are also extremely high, obtaining a value of 0.99, so they are satisfied in most of the transactions.

Finally, we have applied the classic Apriori algorithm to the same dataset in order to demonstrate that the results and the information described by an association rule on multiple instance data is quite different to the information provided by classic association rules. The rule IF (amountlcy = [-172,034.22, max] AND currency = LTL) THEN (productCategory = current account) is satisfied by 88.3% of the transactions when the data is organized into multiple instances where each transaction is a different client and each instance is a single banking operation. On the contrary, using a classic data representation where each transaction is a different banking operation, the aforementioned rule is satisfied by 68% of the transactions, which is quite different to the support obtained by MI-ARM. Additionally, the rule IF (currency = LTL) THEN (productCategory = current account) is really accurate in MI-ARs, since 94.2% of the clients whose operations involve Lithuanian litas (LTL) are related to the current account of the clients. Nevertheless, if we analyse the same rule in classic ARM, we obtain an accuracy of 75%, since the information represented is completely different as previously described in Section 5.1.

6.2. Application to the Spanish public employment service

In this section, we also apply the new problem formulation to another real-world problem in accordance with the Spanish public employment service. Data comprises information about the unemployed in Córdoba, Spain.

Data description. Data have been gathered from the Spanish public employment service, which has an automatic system that enables records of the unemployed to be recognised. This system, known as SARE, was designed and implemented by the computing service located in Córdoba, Spain. The aim of the SARE system is to inform the Spanish citizens about the unemployment benefits that they could receive from the State party. Data were gathered from 169 working days, from January 2014 to September 2014, and comprises information about 63,185 citizens having 133 attributes each one.

Data have been organized into transactions that represent different regions. Each transaction comprises a different number of instances, one per unemployed. This data organization enables MI-ARM to be applied on it, providing a description based on the regions' information in such a way that those regions with a lower number of unemployed are as important as others with a higher number. Finally, data have also been organized into transactions that represent people from different ages, so the results are not biased against ages with a lower rate of unemployed.

Results. Applying MI-ARM to the aforementioned dataset we obtained a series of rules with different grades of accuracy (known as confidence in this field). Table 6 shows the MI-ARs obtained from data organized into regions, depicting the support, confidence and lift values. It should be noted that a pattern is satisfied in a specific transaction if and only if at least one instance within the transaction is satisfied.

The two first rules show the relation between the gender and the number of working days (considering all the regions as transactions of the dataset). As described by the confidence quality measure, only 25% of the women who are close to the retirement age have been working for more than a year. On the contrary, 100% of the men who are close to the retirement age have been working for more than a year. It shows a reality in Spain, where the fact of obtaining a job is easier for men than for women, specially when they are older than 50. Continuing the analysis, the following three rules determine how important the age is to obtain a job. They determine that, regardless the gender, the fact of being working for more than a year is much more probable if a person is younger than 30, i.e. an accuracy of 66.6%, than if a person is older than 51 — only 25% of the people older than this age have been working more than a year. It should be noted that all these rules were obtained from people who have requested for a unemployment benefit.

Continuing the analysis, the two last rules shown in Table 6 are analysed together with the rules shown in Table 7. These two rules show the relationship between the gender and the number of working days without any distinction between the age of the worker. It should be noted that, whereas the rules depicted in Table 6 were obtained considering the transactions or-

Table 6: Multiple-instance association rules obtained from data of the Spanish public employment service. Transactions organized by regions. Support and confidence are represented in per unit basis.

Rule	**Support**	**Confidence**	**Lift**
IF (age = [51, 65] AND gender = female) THEN (Number of working days = [1 year, max])	0.111	0.250	0.562
IF (age = [51, 65] AND gender = male) THEN (Number of working days = [1 year, max])	0.111	1.000	2.250
IF (age = [min, 30]) THEN (Number of working days = [1 year, max])	0.222	0.666	1.500
IF (age = [31, 50]) THEN (Number of working days = [1 year, max])	0.333	0.375	0.844
IF (age = [51, max]) THEN (Number of working days = [1 year, max])	0.111	0.250	0.562
IF (gender = female) THEN (Number of working days = [1 year, max])	1.000	1.000	1.000
IF (gender = male) THEN (Number of working days = [1 year, max])	1.000	1.000	1.000

ganized by regions, the rules depicted in Table 7 were obtained considering the transactions organized by ages. If we compare both, it is stated that the rules are much more accurate, i.e. a reliability of 100%, when the transactions are organized by regions than when the transactions are organized by ages, i.e. reliabilities lower than 40%. This difference shows the usefulness of applying multiple-instance in the field of association rule mining, obtaining different perspectives depending on the data organization. Considering the aforementioned rules, in any Spanish region it is possible to find at least a man or a woman who has been working for more than a year. On the contrary, considering that the transactions are organized by ages, this assertion

Table 7: Multiple-instance association rules obtained from data of the Spanish public employment service. Transactions organized by ages. Support and confidence are represented in per unit basis.

Rule	**Support**	**Confidence**	**Lift**
IF (gender = female) THEN (Number of working days = [1 year, max])	0.333	0.375	0.844
IF (gender = male) THEN (Number of working days = [1 year, max])	0.222	0.333	0.750

is not true, since only 33% of the ages comprise at least a person who have been working for more than a year. This analysis is quite interesting, since it shows how important is the working age to obtain a position in a company.

Finally, we have applied the classic Apriori algorithm to the same dataset in order to demonstrate that the results and the information provided are different to the obtained by classic association rules. The rule IF (age = [31, 50]) THEN (Number of working days = [1 year, max]) is satisfied by 33.3% of the transactions when the data is organized into multiple instances where each transaction is a different region and each instance is an unemployed. On the contrary, using a classic data representation where each transaction is a single unemployed, the aforementioned rule is satisfied only by 3.06% of the transactions, which is quite different to the support obtained by MI-ARM.

Additionally, the following two rules were previously analysed by using two different perspectives (Tables 6 and 7): (1) IF (gender = female) THEN (Number of working days = [1 year, max]); (2) IF (gender = male) THEN (Number of working days = [1 year, max]). Analysing these two association rules by a classic methodology, we discover that the first rule is satisfied in 1.92% of the transactions, and the second one in 5.95% of the transactions. These results are completely different to the ones obtained when the rules are applied to multiple instance data, and the meaning is quite different too. For instance, the rule IF (gender = female) THEN (Number of working days = [1 year, max]) applied to the whole dataset represents a relationship between "*gender = female*" y "*Number of working days = [1 year, max]*" for all the unemployed. Nevertheless, if we describe this rule for a multiple instance point of view where transactions are organized by regions (Table 6) or ages (Table 7), the meaning is referred to regions or ages, not to individuals.

7. Concluding remarks

The problem of multiple-instance learning, where data are grouped into transactions and each transaction comprises a set of instances, has been dealt by different researchers from a supervised learning point of view. Nevertheless, multiple-instance data is more and more common so the analysis and description of the behaviour of this type of data is a dare. In this sense we have proposed the concept of multiple-instance association rule mining, which is a new problem that enables descriptions in multiple-instance data to be obtained.

In this paper, we have formally presented the new concept of multiple-instance association rule mining, which could be approached from different perspectives. Nevertheless, for the sake of describing and applying this new concept, we have considered the analysis of an exhaustive search algorithm that is based on the well-known Apriori algorithm in the ARM field. Additionally, we have applied the new problem to two real-world fields, obtaining interesting relationships between patterns in data inherently multi-instance.

Authors are aware that this new problem could be considered from different perspectives not considered in this work or even in classic ARM yet. Algorithms and formulations described in this paper serve as a starting point for further researches.

Acknowledgements

This research was supported by the Spanish Ministry of Economy and Competitiveness, project TIN-2014-55252-P, and by FEDER funds.

References

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of 20th International Conference on Very Large Data Bases*, VLDB'94, pages 487–499, Santiago de Chile, Chile, 1994. Morgan Kaufmann.

[2] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *Advances in Neural Information Processing Systems*, pages 561–568. MIT Press, 2003.

[3] F. Berzal, I. Blanco, D. Sánchez, and M. A. Vila. Measuring the accuracy and interest of association rules: A new framework. *Intelligent Data Analysis*, 6(3):221–235, 2002.

[4] L. Cagliero, S. Chiusano, P. Garza, and G. Bruno. Pattern set mining with schema-based constraint. *Knowledge-Based Systems*, 84:224 – 238, 2015.

[5] J. Chevaleyre and D. Zucker. Learning rules from multiple instance data: issues and algorithms. In *Proceeding of the 9th Information Processing and Management of Uncertainty in Knowledge-Based Systems*, IPMU, pages 455–459, Annecy, France, 2002.

[6] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Perez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31 – 71, 1997.

[7] J. Han, J. Pei, Y. Yin, and R. Mao. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data Mining and Knowledge Discovery*, 8:53–87, 2004.

[8] M. Hassan, A. Karim, J.-B. Kim, and M. Jeon. CDIM: Document clustering by discrimination information maximization. *Information Sciences*, 316:87–106, 2015.

[9] H. Liu, S. Zhang, and X. Wu. Mlslr: Multilabel learning via sparse logistic regression. *Information Sciences*, 281:310–320, 2014.

[10] J. M. Luna, A. Cano, M. Pechenizkiy, and S. Ventura. Speeding-Up Association Rule Mining With Inverted Index Compression. *IEEE Transactions on Cybernetics*, pp(99):1–14, 2016.

[11] J. M. Luna, C. Romero, J. R. Romero, and S. Ventura. An evolutionary algorithm for the discovery of rare class association rules in learning management systems. *Applied Intelligence*, 42(3):501–513, 2015.

[12] J. M. Luna, J. Romero, and S. Ventura. On the adaptability of G3PARM to the extraction of rare association rules. *Knowledge and Information Systems*, 38(2):391–418, 2014.

[13] J. M. Luna, J. R. Romero, C. Romero, and S. Ventura. On the use of genetic programming for mining comprehensible rules in subgroup discovery. *IEEE Transactions on Cybernetics*, 44(12):2329–2341, 2014.

[14] J. M. Luna, J. R. Romero, C. Romero, and S. Ventura. Reducing gaps in quantitative association rules: a genetic programming free-parameter algorithm. *Integrated Computer Aided Engineering*, 21(4):321–337, 2014.

[15] O. Maron and T. Lozano-Perez. A framework for multiple-instance learning. In *Advances in Neural Information Processing Systems*, pages 570–576. MIT Press, 1998.

[16] E. Ng, A. W.-C. Fu, and K. Wang. Mining association rules from stars. In *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002)*, pages 322–329, Maebashi City, Japan, 2002.

[17] C. Ordoñez, N. Ezquerra, and C. Santana. Constraining and summarizing association rules in medical data. *Knowledge and Information Systems*, 9(3):259–283, 2006.

[18] A. K. H. Tung, H. Lu, J. Han, and L. Feng. Efficient mining of intertransaction association rules. *IEEE Transactions on Knowledge and Data Engineering*, 15(1):43–56, 2003.

[19] J. Wang. Solving the multiple-instance problem: a lazy learning approach. In *Proceeding of the 17th International Conference on Machine Learning*, ICML, pages 1119–1125, Standford, CA, USA, 2000. Morgan Kaufmann.

[20] X. Yan, C. Zhang, and S. Zhang. ARMGA: Identifying interesting association rules with genetic algorithms. *Applied Artificial Intelligence*, 19(7):677–689, 2005.

[21] A. Zafra and S. Ventura. Multi-instance genetic programming for predicting student performance in web based educational environments. *Applied Soft Computing*, 12(8):2693 – 2706, 2012.

[22] X. Zhang and Z. Deng. Mining summarization of high utility itemsets. *Knowledge-Based Systems*, 84(0):67 – 77, 2015.

[23] Z. Zhou, K. Jiang, and M. Li. Multi-instance learning based web mining. *Applied Intelligence*, 22(2):135–147, 2005.

[24] J. Zucker and Y. Chevaleyre. Solving multiple-instance and multiple-part learning problems with decision trees and decision rules. Application to the mutagenesis problem. In *Proceedings of the 14th Biennial Conference of the Canadian Society on Computational Studies of Intelligence: Advances in Artificial Intelligence*, pages 204–214, 2001.

[25] J. Zucker and J. Ganascia. Changes of representation for efficient learning in structural domains. In *Proceedings of the 13th International Conference on Machine Learning (ICML 1996)*, pages 543–551, Bary, Italy, 1996.

[26] J. Zucker and J. Ganascia. Learning structurally indeterminate clauses. In *Proceedings of the 8th International Workshop on Inductive Logic Programming (ILP 1998)*, pages 235–244, Berlin, Germany, 1998.