An Algorithm for the Discovery of Arbitrary Length Ordinal Association Rules

Alina Campan, Gabriela Serban, Traian Marius Truta, and Andrian Marcus

***Abstract*—Association rule mining techniques are used to search attribute-value pairs that occur frequently together in a data set. Ordinal association rules are a particular type of association rules that describe orderings between attributes that commonly occur over a data set [9]. Although ordinal association rules are defined between any number of the attributes, only discovery algorithms of binary ordinal association rules (i.e., rules between two attributes) exist.**

In this paper, we introduce the DOAR algorithm that efficiently finds all ordinal association rules of interest to the user, of any length, which hold over a data set. We present a theoretical validation of the algorithm and experimental results obtained by applying this algorithm on a real data set.

I. Introduction

Association rule mining aims to find interesting associations or correlations that exist between items in large data sets. Association rule discovery was first introduced in the context of market basket analysis, where customer buying habits or patterns are to be uncovered [2]. Since then, many research efforts in the area of association rule mining have been made mainly in two directions:

- To improve old algorithms or develop new ones in order to ensure scalability with respect to data size [10] [6].
- To extend the Boolean association rules concept to adapt it to new applications. Han and Kamber [5] present an extensive overview of the types of association rules that can be discovered in data (e.g., Boolean vs. quantitative, single vs. multi-dimensional, single vs. multi-level, constrained-based rules, etc.) and of their utility and discovery methods.

A. Campan is with the Department of Computer Science, Babes-Bolyai University, Cluj-Napoca, Romania (phone: +40-746-881690; e-mail: alina@ cs.ubbcluj.ro).

G. Serban is with the Department of Computer Science, Babes-Bolyai University, Cluj-Napoca, Romania (e-mail: gabis@cs.ubbcluj.ro).

T. M. Truta is with the Department of Computer Science, Northern Kentucky University, USA (e-mail: trutat1@nku.edu).

A. Marcus is with the Department of Computer Science, Wayne State University, USA (e-mail: amarcus@wayne.edu). He is currently visiting in the Department of Computer Science, Babes-Bolyai University, Cluj-Napoca, Romania.

Within the second direction of research, a new kind of association rules, ordinal association rules (a.k.a. ordinal rules), was introduced in [9]. Given a set of records, described by a set of attributes, the ordinal association rules identify ordinal relationships between the attribute values that hold for a certain percentage of the records. There are several existing and potential applications for ordinal association rules, such as automatic detection of errors in data sets [8].

Although ordinal association rules are defined between any number of attributes, discovery algorithms exist only for binary ordinal association rules (i.e., rules between two attributes) [9].

In this paper, we introduce an algorithm that efficiently finds all ordinal association rules of any length (i.e., between multiple attributes) that hold over a data set, and which are of interest to the user.

The paper is structured as follows. Section II presents the formal definition of the ordinal association rules. Section III introduces and explains the DOAR algorithm for uncovering all the interesting ordinal rules in a data set. Theoretical validation of the algorithm is given in Section IV. Section V presents a case study on a real data set that shows the algorithm's capacity in reducing the search space for ordinal rules. Conclusions and future work are outlined in Section VI.

II. Ordinal Association Rules

Datasets that contain several attributes with similar or comparable domains of values are frequent in data mining. The order relationships between record attributes that hold for a certain percentage of records represent an extension of association rules and they are called ordinal association rules [9].

Definition 1. [9] Let $R = \{r_1, r_2, \dots, r_n\}$ be a set of records, where each record is a set of m attributes, $(a_1,\dots, a_m)$. We denote by $\Phi(r_j, a_i)$ the value of attribute a_i in the record r_j. Each attribute a_i takes values from a domain D, which also contains ε (empty, null). The following relations (partial orderings) are defined over domain D: less or equal ($\leq$), equal ($=$), greater or equal ($\geq$), all having the usual meaning. An **ordinal association rule** is an expression of the form

$(a_{i_1}, a_{i_2}, \ldots, a_{i_\ell}) \Rightarrow (a_{i_1}\ \mu_1\ a_{i_2}\ \ldots \mu_{\ell-1}\ a_{i_\ell})$, where

$\{a_{i_1}, a_{i_2}, \ldots, a_{i_\ell}\} \subseteq A = \{a_1, a_2 \ldots, a_m\}$,

$a_{i_j} \neq a_{i_k}, \forall j, k = 1..\ell, j \neq k$, and

$\mu_i \in M = \{\leq, =, \geq\}$. If:

- $a_{i_1}, a_{i_2}, \ldots, a_{i_\ell}$ occur together (are non-empty) in s% of the n records then we call s the *support* of the rule;

and

- we denote by $R' \subseteq R$ the set of records where $a_{i_1}, a_{i_2}, \ldots, a_{i_\ell}$ occur together and $\phi(r_j, a_{i_1})\ \mu_1\ \phi(r_j, a_{i_2}) \ldots \mu_{\ell-1}\ \phi(r_j, a_{i_\ell})$ is true for each record r_j in R', then $c = |R'| / |R|$ is called the *confidence* of the rule.

The users usually need to uncover interesting ordinal association rules that hold in a data set; they are interested in rules which hold between a minimum number of records, that is rules with support at least *min_s* and confidence at least *min_c* (*min_s* and *min_c* are user-provided thresholds).

Definition 2. We call an ordinal association rule in R **interesting** if its support s is greater than or equal to a user-specified minimum support, *min_s* and its confidence c is greater than or equal to a user-specified minimum confidence, *min_c*.

We introduce a new concept, necessary for the definition of our novel discovery algorithm.

Definition 3. The **length,** ℓ, of an ordinal association rule $(a_{i_1}, a_{i_2}, \ldots, a_{i_\ell}) \Rightarrow (a_{i_1}\ \mu_1\ a_{i_2}\ \ldots \mu_{\ell-1}\ a_{i_\ell})$ is the number of attributes in the rule.

Previous work [9] proposed an identification process for the binary ordinal association rules (i.e., rules having the length 2) that have confidence greater than a given threshold.

III. Discovery of Ordinal Association Rules - DOAR

We introduce a new algorithm, called DOAR (Discovery of Ordinal Association Rules), to discover all the *interesting* (w.r.t. the user-specified thresholds *min_s* and *min_c*) ordinal rules of *any length* in a data set. Our algorithm is inspired by the Apriori algorithm [3] for determining Boolean association rules in a transactional data set. Namely, rules identification is an iterative process that consists in length-level generation of candidate rules, followed by the verification of the candidates for minimum support and confidence compliance.

The DOAR algorithm performs multiple passes over the data set R. In the first pass, it calculates the support and confidence of the 2-length rules and determines which of them are interesting, i.e., verify minimum support and confidence requirement. In every subsequent pass over the data, we start with a seed set of interesting rules, found in the previous pass. We use this set to generate new possible interesting rules, called *candidate rules*, and we compute the actual support and confidence of these candidates during the scan of the data. At the end of this step, we keep the rules that are deemed interesting, which will be used in the next iteration. The process stops when no new interesting rules were found in the latest iteration.

The remainder of this section explains in details and formalizes the main steps of the algorithm, discusses the complexity of the algorithm, and provides a usage example.

A. The DOAR Algorithm

DOAR makes use of the following sets:

- C_k is the set of k-length candidate rules; a k-length candidate rule is a sequence of partial orderings between k attributes, $2 \leq k \leq m$;
- L_k is the set of the k-length interesting (i.e., support and confidence greater than or equal with *min_s* and *min_c*, respectively) ordinal rules found by DOAR. It will be proved that L_k is equal to the set of all k-length interesting ordinal association rules existing in data, $2 \leq k \leq m$.

The DOAR algorithm starts by generating C_2, computing the support and confidence for each candidate rule in C_2, and determining L_2. For the set $M = \{\leq, =, \geq\}$ of partial ordering relations between attributes, the binary candidate rules (C_2) are generated as specified in line 1 of the algorithm (see Fig. 1). The L_2 set is determined by a scan of the data and is the starting point of the subsequent steps in the iterative process employed by DOAR.

Every iteration consists of two phases:

- First, DOAR generates the k-length candidate rules set, C_k ($k \geq 3$), using the set of (k-1)-length interesting rules, L_{k-1}. The candidate generation process is the key element of our algorithm.
- Then, a scan of the R data set is performed, while computing the support and the confidence of every candidate rule in C_k. The candidates in C_k that have minimum support and satisfy the confidence requirements are interesting ordinal association

rules and therefore are included in L_k.

At every iteration, candidates are generated by the *GenCandidates* function (see Fig. 1). The *GenCandidates* function has as argument the L_{k-1} set of (k-1)-length interesting rules and returns C_k , a superset of the set of the interesting k-length rules. The elements of C_k are sequences of partial orderings between k attributes, called candidate k-length rules. *GenCandidates* produces the candidates in C_k in the following manner. Each unordered pair of rules ($rule_1$, $rule_2$), $rule_1$, $rule_2 \in L_{k-1}$, which satisfies one of the formats below, is merged into a candidate rule c. To simplify the notation in these formulas, we only write from each rule the partial orderings sequence (i.e., the right hand side of the rule). We mention that $a^1, a^2, a_{i_1}, a_{i_2}, \ldots, a_{i_{k-2}} \in A$ are attributes, $\mu^1, \mu^2, \mu_1, \ldots, \mu_{k-3} \in M$ are relations and μ^{-1} denotes the converse of the relation $\mu \in M$.

$rule_1 \equiv (a^1\ \mu^1\ \mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}})$ and

$$rule_2 \equiv (\mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}}\ \mu^2\ a^2), \tag{1}$$

then $c \equiv (a^1\ \mu^1\ \mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}}\ \mu^2\ a^2)$,

or

$rule_1 \equiv (\mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}}\ \mu^1\ a^1)$ and

$$rule_2 \equiv (a^2 \mu^2\ \mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}}), \tag{2}$$

then $c \equiv (a^2 \mu^2\ \mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}}\ \mu^1\ a^1)$,

or

$rule_1 \equiv (a^1\ \mu^1\ \mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}})$ and

$$rule_2 \equiv (a^2\ \mu^2\ \mathbf{a_{i_{k-2}}}\ \boldsymbol{\mu_{k-3}^{-1}} \ldots \mathbf{a_{i_2}}\ \boldsymbol{\mu_1^{-1}}\ \mathbf{a_{i_1}}), \tag{3}$$

then $c \equiv (a^1 \mu^1\ \mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}}\ (\mu^2)^{-1}\ a^2)$,

or

$rule_1 \equiv (\mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}}\ \mu^1\ a^1)$ and

$$rule_2 \equiv (\mathbf{a_{i_{k-2}}}\ \boldsymbol{\mu_{k-3}^{-1}} \ldots \mathbf{a_{i_2}}\ \boldsymbol{\mu_1^{-1}}\ \mathbf{a_{i_1}}\ \mu^2\ a^2), \tag{4}$$

then $c \equiv (a^2\ (\mu^2)^{-1}\ \mathbf{a_{i_1}}\ \boldsymbol{\mu_1}\ \mathbf{a_{i_2}} \ldots \boldsymbol{\mu_{k-3}}\ \mathbf{a_{i_{k-2}}}\ \mu^1\ a^1)$.

The semantics of these formulas is explained in Section IV.

Fig. 1 shows the pseudo-code version of the DOAR algorithm for generating all the interesting ordinal association rules that hold over a data set R.

In Section IV we prove the completeness of the DOAR algorithm.

B. *Asymptotic Analysis*

The discovery of interesting ordinal rules that hold over a data set is, in fact, a search problem. The brute force method (i.e., the "generate and test" method) for solving

```
Algorithm DOAR is
// Input: data set R, min_s, min_c;
// Output: the set Answer of all interesting ordinal association rules that hold over R.
    C2 ={(ai1,ai2)⇒(ai1 μ1 ai2) | ai1,ai2 ∈A, i1,i2 =1..m,i1 <i2, μ1 ∈M};
    Scan R and compute the support and confidence of candidates in C2;
    Keep the interesting rules from C2 ⇒ L2;
    k = 3;
    While (Lk-1 ≠ ∅ and k<=m) do
         Ck = GenCandidates(Lk-1);
         Scan R and compute the support and confidence of candidates in Ck;
         Keep the interesting rules from Ck ⇒ Lk;
         k = k + 1;
    End;
    Answer = ⋃_k Lk ;
EndDOAR.
```

Fig. 1. Algorithm for the Discovery of Arbitrary Length Ordinal Association Rules (DOAR)

this problem consists in generating and verifying for support and confidence all possible interesting ordinal association rules, i.e., all sequences of partial orderings between k attributes, $2 \le k \le m$. This set is exponential on the number of record attributes (m).

The DOAR algorithm significantly prunes the exponential search space of all possible interesting ordinal association rules, due to the candidate generation technique. The candidate generation restricts the search to those regions of the search space where is possible that interesting rules exist. It prunes out all the regions where is impossible to find any interesting rule. The search space reduction depends on the data being analyzed. The larger the number of interesting rules in the data set is, the larger the size of the candidates sets will be. In addition, the number of data set scans grows with the length of the interesting rules in the data set.

In a worst case scenario, the overall time complexity of the merge operations (i.e., rules (1)-(4), line 6 in the algorithm) is

$$O\left(\sum_{k=3}^{m} k \cdot \left|L_{k-1}\right|^2\right)$$

and the overall time complexity of the candidate verification operations (lines 7 and 8 in the algorithm) is

$$O\left(n \cdot \sum_{k=3}^{m} k \cdot \left|C_k\right|^2\right).$$

C. Example

To better explain the concept of ordinal rules and the DOAR algorithm, we give an example of applying it on a data set sample, R, shown in TABLE 1. The data set is artificially generated and it is composed of integer value data elements, grouped in records.

TABLE 1 THE DATA SET R

	a_1	a_2	a_3	a_4
r_1	2	4	3	1
r_2	5	6	8	7
r_3	9	10	12	11
r_4	12	15	13	11
r_5	1	2	4	3
r_6	5	6	8	7
r_7	9	10	12	11
r_8	12	15	13	16
r_9	27	21	29	24
r_{10}	30	34	29	38

In this example, we are interested in discovering all the ordinal rules with min_s = 90% and min_c = 80%.

In the first step, C_2 is generated as follows:

$$\begin{aligned} \boldsymbol{C_2} = \{ \; & a_1 \le a_2, a_1 = a_2, a_1 \ge a_2, \\ & a_1 \le a_3, a_1 = a_3, a_1 \ge a_3, \\ & a_1 \le a_4, a_1 = a_4, a_1 \ge a_4, \\ & a_2 \le a_3, a_2 = a_3, a_2 \ge a_3, \\ & a_2 \le a_4, a_2 = a_4, a_2 \ge a_4, \\ & a_3 \le a_4, a_3 = a_4, a_3 \ge a_4 \; \}. \end{aligned}$$

By scanning the data set R, only the following 2-length candidate rules were found to be interesting (i.e., respecting the minimum support and confidence condition):

$$\boldsymbol{L_2} = \{a_1 \le a_2, a_1 \le a_3, a_2 \le a_4, a_3 \ge a_4\}.$$

We applied the merge formulas (1)-(4) on the set of 2-length interesting rules, L_2, and we obtained the following set, C_3, of 3-length candidate rules:

$$\begin{aligned} \boldsymbol{C_3} = \{ & a_2 \ge a_1 \le a_3, a_1 \le a_2 \le a_4, \\ & a_1 \le a_3 \ge a_4, a_2 \le a_4 \le a_3\}. \end{aligned}$$

From these candidate rules, only two verify the minimum support and confidence requirement. These two rules form the set L_3, given below:

$$\boldsymbol{L_3} = \{a_2 \ge a_1 \le a_3, a_1 \le a_3 \ge a_4\}.$$

There exist one 4-length candidate rule, but this candidate is not interesting, its confidence is only 70%. The C_4 and L_4 sets are given below.

$$\boldsymbol{C_4} = \{a_2 \ge a_1 \le a_3 \ge a_4\}, \; \boldsymbol{L_4} = \varnothing.$$

As in the last step no new interesting rules were found, the process stops.

In this example, the search for interesting rules would stop anyway, as it reached the maximum possible length for a rule (i.e., the number of record attributes).

IV. THEORETICAL VALIDATION

We prove the completeness of the DOAR algorithm for generating and verifying candidate rules – namely, no interesting ordinal rules that exist in the data can be missed by this process.

We also show that no redundant ordinal rules are generated through the DOAR algorithm. We achieve this by proving that the starting set C_2 does not contain redundant ordinal rules and neither the subsequent steps in the process will not produce such rules.

A. Construction of C_2 and L_2

We examine the construction of the C_2 set. For the set $M = \{\le, =, \ge\}$ of partial ordering relations between attributes, the binary candidate rules (C_2) are generated as specified in line 1 of the algorithm (see Fig. 1).

Lemma 1. It is not necessary to consider as candidate rules all the partial orderings between all ordered pairs of attributes in A, i.e., $\{(a_{i_1}, a_{i_2}) \Rightarrow (a_{i_1}\ \mu_1\ a_{i_2}) \mid a_{i_1}, a_{i_2} \in A$, $a_{i_1} \neq a_{i_2}$, $\mu_1 \in M\}$.

Proof:
$\forall$ $\mu \in M$, its converse $\mu^{-1} \in M$. So, if $(a_{i_1}, a_{i_2}) \Rightarrow (a_{i_1}\ \mu_1\ a_{i_2})$ is an interesting rule, then $(a_{i_2}, a_{i_1}) \Rightarrow (a_{i_2}\ \mu_1^{-1}\ a_{i_1})$ is also interesting. It suffices to verify one of these two orderings for support and confidence in order to decide if they both define interesting rules or not; verifying both these converse binary expressions would be redundant.

The C_2 candidate rules that have support and confidence greater than their given thresholds (*min_s* and *min_c*) are included in the L_2 set. By limiting the L_2 seed set in this way, the algorithm will avoid generating (verifying) converse candidates (rules), at any higher length level.

B. Candidate Generation

To explain the procedure for candidate construction and to prove its completeness, we introduce the concept of binary ordinal rules graph, as defined below.

Definition 4. Given the L_2 set of binary interesting ordinal rules, the **binary ordinal association rules graph,** G_2 is an oriented graph defined as follows: $G_2 = (A,E)$, where:

- The set A of vertices is the set of record attributes.
- The set E of edges is $E = \{(a_i, a_j)_\mu \mid$ if there exist a binary rule $(a_i, a_j) \Rightarrow (a_i\ \mu\ a_j)$ or $(a_j, a_i) \Rightarrow (a_j\ \mu^{-1}\ a_i) \in L_2\} \subseteq A \times A$. μ is called the label of the edge $(a_i, a_j)_\mu$.

Theorem 1 below shows that each interesting ordinal rule that holds over R has a corresponding path in G_2. The converse is not true: not every (elementary) path in G_2 corresponds to an interesting rule.

Theorem 1. If $(a_{i_1}, a_{i_2}, a_{i_3}, \ldots, a_{i_k}) \Rightarrow (a_{i_1}\ \mu_1\ a_{i_2}\ \mu_2\ a_{i_3} \ldots \mu_{k-1} a_{i_k})$ is a k-length interesting ordinal association rule, then a path $\{(a_{i_1}, a_{i_2})_{\mu_1}, (a_{i_2}, a_{i_3})_{\mu_2}, \ldots, (a_{i_{k-1}}, a_{i_k})_{\mu_{k-1}}\}$ exists in G_2.

Proof:
If $(a_{i_1}, a_{i_2}, a_{i_3}, \ldots, a_{i_k}) \Rightarrow (a_{i_1}\ \mu_1\ a_{i_2}\ \mu_2\ a_{i_3} \ldots \mu_{k-1}\ a_{i_k})$ is an interesting ordinal association rule over R, denoted by r, then it satisfies the minimum support and confidence requirement. This means that:

- $a_{i_1}, a_{i_2}, a_{i_3}, \ldots, a_{i_k}$ occur *together* in at least *min_s*% of the n records in R $\Rightarrow$ $\forall j, j = 1..k-1$, a_{i_j} and $a_{i_{j+1}}$ also occur together in at least *min_s*% of the n records. Therefore, the support of the rule $(a_{i_j}, a_{i_{j+1}}) \Rightarrow (a_{i_j}\ \mu_j\ a_{i_{j+1}})$ is greater or at least equal to the support of the ordinal rule r.

and

- if $R' \subseteq R$ is the set of all the records where $a_{i_1}, a_{i_2}, a_{i_3}, \ldots, a_{i_k}$ occur together and $\phi(r, a_{i_1})\ \mu_1\ \phi(r, a_{i_2}) \ldots \mu_{k-1}\ \phi(r, a_{i_k})$ is true for each record r in R', then $|R'|/|R| \geq$ *min_c*. In this case, if we denote by $R'' \subseteq R$ the set of all records where a_{i_j} and $a_{i_{j+1}}$ occur together and $\phi(r, a_{i_j}) \mu_j \phi(r, a_{i_{j+1}})$ is true for each record r in R'', then $R' \subseteq R''$. So, $|R''|/|R| \geq |R'|/|R| \geq$ *min_c*, $\forall j, j = 1..k-1$.

It follows that, if r is an interesting ordinal rule over R, then $(a_{i_j}, a_{i_{j+1}}) \Rightarrow (a_{i_j}\ \mu_j\ a_{i_{j+1}})$ are all interesting rules in R, $\forall j$, j=1..k-1, as they satisfy the minimum support and confidence requirement. Hence, L_2 contains either rule $(a_{i_j}, a_{i_{j+1}}) \Rightarrow (a_{i_j}\ \mu_j\ a_{i_{j+1}})$ or rule $(a_{i_{j+1}}, a_{i_j}) \Rightarrow (a_{i_{j+1}}\ \mu_j^{-1}\ a_{i_j})$. So, according to the definition of the graph G_2, there exist the edges $(a_{i_j}, a_{i_{j+1}})_{\mu_j}$, $\forall j$, j=1..k-1. So, it exists in G_2 the path $\{(a_{i_1}, a_{i_2})_{\mu_1}, (a_{i_2}, a_{i_3})_{\mu_2}, \ldots, (a_{i_{k-1}}, a_{i_k})_{\mu_{k-1}}\}$ that corresponds to the interesting ordinal rule r.

Note: As an ordinal rule contains distinct attributes, their corresponding paths in G_2 are elementary (the path vertices are distinct).

In the following we enounce and prove a second theorem, needed to explain the semantic of the formulas (3) and (4) for joining pairs of rules to generate candidates.

Theorem 2. If there is a path $\{(a_{i_1}, a_{i_2})_{\mu_1}, (a_{i_2}, a_{i_3})_{\mu_2}, \ldots, (a_{i_{k-1}}, a_{i_k})_{\mu_{k-1}}\}$ in G_2, then there also exists in G_2 the "reverse" path $\{(a_{i_k}, a_{i_{k-1}})_{(\mu_{k-1})^{-1}}, \ldots, (a_{i_3}, a_{i_2})_{\mu_2^{-1}}, (a_{i_2}, a_{i_1})_{\mu_1^{-1}}\}$.

Proof:

If there is an edge $(a_{i_j}, a_{i_{j+1}})_{\mu_j}$ in G_2, then, according to the definition of G_2, there is in L_2 a rule $(a_{i_j}, a_{i_{j+1}}) \Rightarrow (a_{i_j}\ \mu_j\ a_{i_{j+1}})$ or $(a_{i_{j+1}}, a_{i_j}) \Rightarrow (a_{i_{j+1}}\ \mu_j^{-1}\ a_{i_j})$. Either way, it follows that the graph G_2 contains an edge $(a_{i_{j+1}}, a_{i_j})_{\mu_j^{-1}}$ $\forall j$=1..k-1. So, the existence of the reverse path in G_2 is proved.

The semantics of the four join formulas is now clear on the basis of the previous two theorems. Each of these joins reunites two (k-1)-length paths in G_2 (interesting rules in L_{k-1}) that share a (k-2)-length sub-path – formulas (1) and (2), or two (k-2)-length converse sub-paths – formulas (3) and (4). We need to consider all these four join-cases in order not to produce converse candidates (rules) in C_k (L_k), at any k-length level. For example, it does not make sense to check and report as interesting rules both $a_2 \geq a_1 \leq a_3$ and $a_3 \geq a_1 \leq a_2$. Having such converse candidates (rules) would imply wasteful processing and would produce redundant equivalent rules.

Now we can prove the completeness of the candidate rules generation procedure. We need to show that $C_k \supseteq L_k$. Obviously, for every interesting ordinal rule $(a_{i_1}, a_{i_2}, \ldots, a_{i_k}) \Rightarrow (a_{i_1}\ \mu_1\ a_{i_2} \ldots \mu_{k-1}\ a_{i_k})$, each of its sub expressions of the form $(a_{i_j}, a_{i_{j+1}}, \ldots, a_{i_{j+s}}) \Rightarrow (a_{i_j}\ \mu_j\ a_{i_{j+1}} \ldots \mu_{j+s-1}\ a_{i_{j+s}})$ $j \geq 1$, $j+s \leq k$ has to also be an interesting rule. Hence, if we extended every (k-1)-length rule in L_{k-1} (its corresponding path in G_2) with a partial ordering (an edge in G_2, in such a way that the obtained path to be elementary), then we would obtain a superset, C_k, of the set L_k of all the k-length interesting rules that exist in R. It is sufficient that the extension to be performed at the extremities of the path (rule), as depicted in Fig. 2.

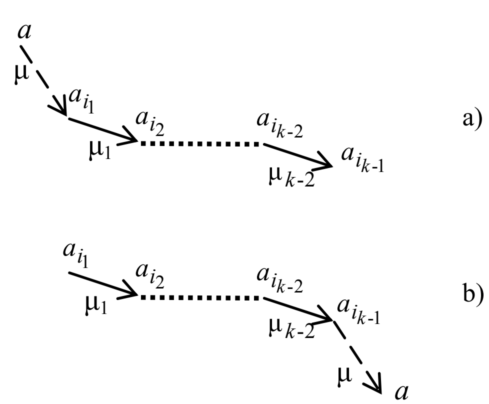

Fig. 2. Candidate rule generation by extension

An extension by insertion, as shown in Fig. 3 (by the means of the edges $(a_{i_j}, a)_{\mu}$ and $(a, a_{i_{j+1}})_{\mu'}$) is redundant. If the path obtained by this insertion represents an interesting rule, then it would be obtained by an extension to the end of another (k-1)-length path, for example the path $\{(a_{i_1}, a_{i_2})_{\mu_1}, \ldots, (a_{i_j}, a)_{\mu}, (a, a_{i_{j+1}})_{\mu'}, \ldots, (a_{i_{k-3}}, a_{i_{k-2}})_{\mu_{k-3}}\}$ in Fig. 3, which passes through the vertices $a_{i_1}, a_{i_2}, \ldots, a_{i_j}$, a, $a_{i_{j+1}}, \ldots, a_{i_{k-2}}$:

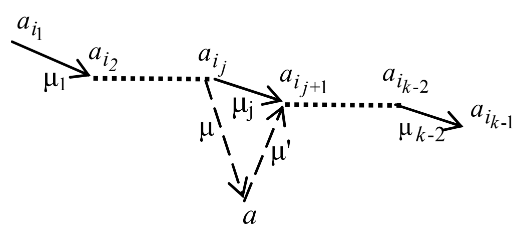

Fig. 3. Candidate rule generation by insertion

Let C_k' be the set of k-length paths (not necessarily interesting rules!), formed by the extension to the end of all the (k-1)-length interesting rules, as described above. Clearly, we can eliminate from C_k' the paths for which the (k-1)-length sub-path, distinct from the rule from which the path was obtained, is not an interesting rule. If we perform such a pruning, then we would still remain with a superset, C_k, of the set L_k. For example, in Fig. 2 (a), such a pruning would be to eliminate the candidate $(a, a_{i_1}, a_{i_2}, \ldots, a_{i_{k-1}}) \Rightarrow (a\ \mu\ a_{i_1}\ \mu_1\ a_{i_2} \ldots \mu_{k-2}\ a_{i_{k-1}})$ obtained from the rule $(a_{i_1}, a_{i_2}, \ldots, a_{i_{k-1}}) \Rightarrow (a_{i_1}\ \mu_1\ a_{i_2} \ldots \mu_{k-2}\ a_{i_{k-1}})$ if the expression $(a, a_{i_1}, a_{i_2}, \ldots, a_{i_{k-2}}) \Rightarrow (a\ \mu\ a_{i_1}\ \mu_1\ a_{i_2} \ldots\ \mu_{k-3}\ a_{i_{k-2}})$ does not represent an interesting rule (i.e., is not in L_{k-1}).

The generation procedure we have proposed is equivalent to: extension to the ends of all the (k-1)-length interesting rules (paths) in L_{k-1}, followed by a pruning step, as explained above. Hence, the rule candidate set, C_k, produced by the *GenCandidates* function, is a superset of L_k, $C_k \supseteq L_k$.

V. Experimental Evaluation

In order to establish how well DOAR prunes the search space of all possible interesting ordinal rules, we performed a case study on a real data set, the Breast Cancer Data [1][4].

The data set used in this case study contains information on the symptoms for cancer patients. In this data set there are 457 records and each record represents a patient. Each patient is described by nine attributes [11]. Each attribute represents a cancer related symptom and has an integer value between 1 and 10.

We applied the DOAR algorithm on this data set, in order to identify all the ordinal association rules, which have the support and confidence at least 90% and 81%, respectively. In other words, the minimum support and confidence thresholds were min_s = 90%, min_c = 81%.

We found 20 interesting binary rules, 35 interesting 3-length rules, and 19 interesting 4-length rules (see TABLE 2). There are no more interesting ordinal rule of higher length in the data set, for the minimum support and confidence thresholds we considered.

The sizes of the sets C_k and L_k, obtained by running DOAR with min_s = 90% and min_c = 81% on the data set, are shown in TABLE 2. For comparison, TABLE 2 contains the sizes of the set of all possible k-length ordinal rules, denoted by SS_k.

TABLE 2. CARDINALITIES OF C_k, L_k, AND SS_k

	k=2	k=3	k=4	k=5
$\|C_k\|$	108	89	42	3
$\|L_k\|$	20	35	19	0
$\|SS_k\|$	108	4536	81648	1224720

With our approach, the exponential search space for finding the interesting ordinal rules, is significantly pruned, as it can be seen in TABLE 2, (see the increase in the size of SS_k). For example, for k=3, we explored only 1.96% of the space of all possible sequences of partial orderings between 3 attributes.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we introduced a novel algorithm for the discovery of interesting any length ordinal association rules in data sets. We formally proved that the proposed algorithm, named DOAR, is complete and we showed through a case study that it efficiently explores the search space of the possible rules.

We are working on extending and improving the research results described in this paper towards:

- Validating the scalability of the DOAR algorithm by conducting experiments on large real data sets.
- Defining ordinal association rules that contain repeating attributes; adapting the proposed technique in order to discover such interesting rules.
- Using the ordinal association rules detection together with supervised learning for medical diagnosis prediction. Preliminary work in this direction is reported in [13].
- Extending ordinal association rules towards relational association rules, i.e., rules between attributes with different data domains and relations not only ordinal between attributes.
- Using ordinal association rules of arbitrary length together with other data mining techniques such as classification or regression to increase the accuracy of the predictive models [7]. Binary association rules are currently used in building predictive models in e-banking services [12].

REFERENCES

[1] ***, "Breast Cancer Data", http://www.cormactech.com/neunet/download.html, 2005.

[2] Agrawal, R., Imielinski, T., and Swami, A., "Mining Association rules between Sets of Items in Large Databases", in Proc. of ACM SIGMOD Int'l Conf. on Management of Data, Washington D.C., 1993, pp. 207-216.

[3] Agrawal, R. and Srikant, R., "Fast Algorithms for Mining Association Rules", in Proc. of The 20th Int'l Conf. on Very Large Databases, Santiago, Chile, 1994.

[4] CorMac Technologies Inc, "Discover the Patterns in Your Data", Date Accessed: December 15, 2005.

[5] Han, J. and Kamber, M., *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2001.

[6] Han, J., Pei, J., and Yin, Y., "Mining frequent patterns without candidate generation", in Proc. of ACM Int'l Conf. on Management of Data, 2000, pp. 1-12.

[7] Hong, S. and Weiss, S., "Advances in predictive model generation for data mining", IBM Research Report RC-21570.

[8] Maletic, J. I. and Marcus, A., "Data Cleansing - A prelude to knowledge discovery", in *Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers*, Maimon, O. a. R., L. Editors, Ed. Springer, 2005, pp. 21-36.

[9] Marcus, A., Maletic, J. I., and Lin, K. I., "Ordinal Association Rules for Error Identification in Data Sets", in Proc. of Tenth Int'l Conf. on Information and Knowledge Management (CIKM 2001), Atlanta, GA, November 3-5 2001, pp. 589-591.

[10] Park, J. S., Chen, M. S., and Yu, P. S., "An Effective Hash-Based Algorithm for Mining Association Rules", in Proc. of ACM SIGMOD Int'l Conf. on Management of Data, 1995, pp. 175-186.

[11] Wolberg, W. and Mangasarian, O. L., "Multisurface method of pattern separation for medical diagnosis applied to breast cytology", in Proc. of National Academy of Sciences, 1990, pp. 9193-9196.

[12] Aggellis, V., and Christodoulakis, D., "Association Rules and Predictive Models for e-Banking Services", in Proc. of 1st Balkan Conf. in Informatics, Tessaloniki, Greece, 2003.

[13] Serban, G, Czibula, I.G., Campan, A., "A Programming Interface For Medical Diagnosis Prediction", Studia Universitatis "Babes-Bolyai", Informatica, LI(1), pag. 21-30, 2006.