Expert Systems with Applications 36 (2009) 11418–11428

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

ILP-based concept discovery in multi-relational data mining

Y. Kavurucu[*], P. Senkul[1], I.H. Toroslu

Department of Computer Engineering, Middle East Technical University, 06531 Ankara, Turkey

ARTICLE INFO

Keywords:
ILP
Data mining
MRDM
Concept discovery
Aggregate predicate

ABSTRACT

Multi-relational data mining has become popular due to the limitations of propositional problem definition in structured domains and the tendency of storing data in relational databases. Several relational knowledge discovery systems have been developed employing various search strategies, heuristics, language pattern limitations and hypothesis evaluation criteria, in order to cope with intractably large search space and to be able to generate high-quality patterns. In this work, an ILP-based concept discovery method, namely Confidence-based Concept Discovery (C^2D), is described in which strong declarative biases and user-defined specifications are relaxed. Moreover, this new method directly works on relational databases. In addition to this, a new confidence-based pruning is used in this technique. We also describe how to define and use aggregate predicates as background knowledge in the proposed method. In order to use aggregate predicates, we show how to handle numerical attributes by using comparison operators on them. Finally, we analyze the effect of incorporating unrelated facts for generating transitive rules on the proposed method. A set of experiments are conducted on real-world problems to test the performance of the proposed method.

1. Introduction

Due to the impracticality of single-table data representation, relational databases are needed to store complex data for real life applications. This has led to development of multi-relational learning systems that are directly applied to relational data (Domingos, 2003; Džeroski, 2003). Relational upgrades of data mining and concept learning systems generally employ first-order predicate logic as representation language for background knowledge and data structures. The learning systems, which induce logical patterns valid for given background knowledge, have been investigated under a research area, called Inductive Logic Programming (ILP) (Muggleton, 1999).

Using logic in data mining is a common technique in the literature (Assche, Vens, Blockeel, & Džeroski, 2006; Dehaspe & Raedt, 1997; Frank, Moser, & Ester, 2007; Knobbe, Siebes, & Marseille, 2002; Lee, Tsai, Wu, & Yang, 2008; Leiva, 2002; Muggleton, 1995; Neville, Jensen, Friedland, & Hay, 2003; Perlich & Provost, 2003; Quinlan, 1990; Srinivasan, 1999; Toroslu & Yetisgen-Yildiz, 2005; Yin, Han, Yang, & Yu, 2004). In this work, we propose a predictive[2] concept learning ILP system, namely Confidence-based Concept Discovery (C^2D), which employs relational association rule mining concepts and techniques. We aimed to overcome some of the limitations of previous systems such as mode declaration, requirement of negative data and discarding aggregate predicates. C^2D utilizes absorption operator of inverse resolution for generalization of concept instances in the presence of background knowledge and refines these general patterns into frequent and strong concept definitions with an Apriori-based specialization operator based on confidence.

A challenging problem of relational concept discovery is dealing with intractably large search space. Several relational knowledge discovery systems have been developed employing various search strategies, heuristics, language pattern limitations and hypothesis evaluation criteria, in order to prune the search space. However, there is a trade-off between pruning the search space and generating high-quality patterns. Therefore, the idea is to balance this trade-off with effective pruning mechanisms. To do this, C^2D utilizes four new pruning mechanisms.

Aggregate functions provide a rich mechanism for expressing the characteristics of the relations having one-to-many relationships among them. Such relationships are common in databases. In concept discovery, conditions on aggregation such as *count* < 10 or *sum* > 100 may define the basic characteristic of a given concept better. For this reason, in this work, we extend the background knowledge with aggregate predicates in order to characterize the structural information that is stored in tables and associations between them.

Aggregate predicates have numeric attributes by their nature. Therefore, in order to add aggregate predicates into the system

[*] Corresponding author.
E-mail addresses: yusuf.kavurucu@ceng.metu.edu.tr (Y. Kavurucu), senkul@ceng.metu.edu.tr (P. Senkul), toroslu@ceng.metu.edu.tr (I.H. Toroslu).
URL: http://www.ceng.metu.edu.tr/karagoz/ (P. Senkul).

[1] Tel.: +90 312 2105518.

[2] In predictive ILP systems, there is a specific target concept to be learned in the light of past experiences.

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.02.100

numeric attribute types should also be handled. Since it is not useful and feasible to define concepts on specific numeric values, in this work, only comparison operators containing numeric attributes are considered in concept discovery.

When the target concept has common attribute types with only some of the background predicates, the rest of the predicates (which are called unrelated relations) can never take part in hypothesis. This prevents the generation of transitive rules through such predicates, which is an important drawback when transitive rules are the only way to describe the target concept. To solve this problem, an optional built-in function is implemented in C^2D which generates transitive rules.

Major contributions of this work can be listed as follows:

1. The main difficulty in relational ILP systems is searching in intractably large hypothesis spaces. In order to cope with this problem, relational ILP systems put strong declarative biases on the semantics of hypotheses. In this work, we aimed to relax the declarative biases in such a way that body clauses may have variables which do not exist in the head predicate. On the other hand, in order to reduce the search space, a confidence-based pruning mechanism is used.
2. Many multi-relational rule induction systems require the user to determine the input–output modes of predicate arguments. Since mode declarations require a high level Prolog and domain knowledge, it is not meaningful to expect such a declaration from an ordinary user. Instead of this, we use the information about relationships between entities in the database if given. Therefore, in this work, the novel user knowledge about domain is not required.
3. Muggleton shows that (Muggleton, 1996), the expected error of an hypothesis according to positive versus all (positive and negative) examples do not have much difference if the number of examples is large enough. In other words, logic programs are learnable with arbitrarily low expected error from only positive examples. As relational databases contain only positive information, a pure multi-relational data mining system based on logic programming could be developed which relies on only positive instances stored as relations. Therefore, the proposed system directly works on relational database, without any requirement of negative instances. If negative instances exist in the database, C^2D can also handle them.
4. The definition of confidence is modified to apply Closed World Assumption (CWA) in relational databases. We introduce type relations to the body of the clauses in order to express CWA.
5. The choice of hypothesis evaluation criteria is an important factor on the quality of the generated patterns. In this work, we used an improved confidence-based hypothesis evaluation criterion, namely *f*-metric, which will be described in the following sections.
6. Only some of the ILP-based classification systems define aggregate predicates in their algorithms. However, better rules (higher coverage and accuracy) can be discovered by using aggregate predicates in the background knowledge. To do this, aggregate predicates are defined in first-order logic and used in C^2D.
7. Numerical attributes are handled in a more efficient way. The clauses having comparison operators on numerical attributes are defined and used in the main algorithm.
8. When the target concept has common attribute types with only some of the background predicates, the rest of the predicates (which are called unrelated relations) can never take part in hypothesis. This prevents the generation of transitive rules through such predicates. In order to remove this drawback, we extended the generalization mechanism in such a way that the indirectly related facts of the target concept instance are added to Apriori lattice to allow transitive rules in the hypothesis.

This paper is organized as follows: Section 2 presents the related work. Section 3 gives an overview of the algorithm of C^2D. Section 4 introduces aggregate predicates and presents aggregation in C^2D. Section 5 gives the basic definitions for transitive rule construction and presents generating transitive rules in C^2D. Finally, Section 6 includes concluding remarks.

2. Related work

In this section, we describe some of the well-known systems related to our system.

FOIL (Quinlan, 1990) is one of the earliest concept discovery systems. It is a top-down relational ILP system, which uses refinement graph in the search process. In FOIL, negative examples are not explicitly provided, they are generated on the basis of CWA.

PROGOL (Muggleton, 1995) is a top-down relational ILP system, which is based on inverse entailment. PROGOL extends clauses by traversing the refinement lattice and reduces the hypothesis space by using a set of mode declarations given by the user, and a most specific clause (also called bottom clause) as the greatest lower bound of the refinement graph. A bottom clause is a maximally specific clause, which covers a positive example and is derived using inverse entailment. PROGOL applies the covering approach and supports learning from positive data as in FOIL.

A Learning Engine for Proposing Hypotheses (ALEPH) (Srinivasan, 1999) is similar to PROGOL, whereas it is possible to apply different search strategies, evaluation functions and refinement operators. It is also possible to define more settings in ALEPH such as minimum confidence and support.

Design of algorithms for frequent pattern discovery has become a popular topic in data mining. Almost all algorithms have the same level-wise search technique known as APRIORI algorithm. The level-wise algorithm is based on a breadth-first search in the lattice spanned by a specialization relation between patterns. WARMR (Dehaspe & Raedt, 1997) is a descriptive ILP system that employs Apriori rule to find frequent queries having the target relation by using support criteria.

The proposed work is similar to ALEPH as both systems produce concept definition from given target. WARMR is another similar work in a sense that, both systems employ Apriori-based searching methods. Unlike ALEPH and WARMR, C^2D does not need input/output mode declarations. It only requires type specifications of the arguments, which already exist together with relational tables corresponding to predicates. Most of the ILP-based systems require negative information, whereas C^2D directly works on databases which have only positive data. Similar to FOIL, negative information is implicitly described according to CWA. Finally, it uses a novel confidence-based hypothesis evaluation criterion and search space pruning method.

ALEPH and WARMR can use indirectly related relations and generate transitive rules only with using strict mode declarations. In C^2D, transitive rules are generated by using indirectly related relations without the guidance of mode declarations. Furthermore, this feature is parametric to increase the efficiency for the applications that do not have unrelated facts.

There are some other works that uses aggregation in multi-relational learning. Crossmine (Yin et al., 2004) is such an ILP based multi-relational classifier that uses TupleID propagation. Multi-relational g-mean decision tree, called Mr.G-Tree (Lee et al., 2008) is proposed to extend the concepts of propagation described in Crossmine by introducing the g-mean Tuple ID propagation