Mining RDF Metadata for Generalized Association Rules: Knowledge Discovery in the Semantic Web Era

Tao Jiang
Nanyang Technological University
Nanyang Avenue, Singapore 639798
jian0006@ntu.edu.sg

Ah-Hwee Tan
Nanyang Technological University
Nanyang Avenue, Singapore 639798
asahtan@ntu.edu.sg

ABSTRACT

In this paper, we present a novel frequent generalized pattern mining algorithm, called *GP-Close*, for mining generalized associations from RDF metadata. To solve the *over-generalization* problem encountered by existing methods, GP-Close employs the notion of *generalization closure* for systematic *over-generalization reduction.*

Categories and Subject Descriptors: H.2.8 [Database Applications]: Data mining

General Terms: Algorithms

Keywords: RDF Mining, Association Rule Mining

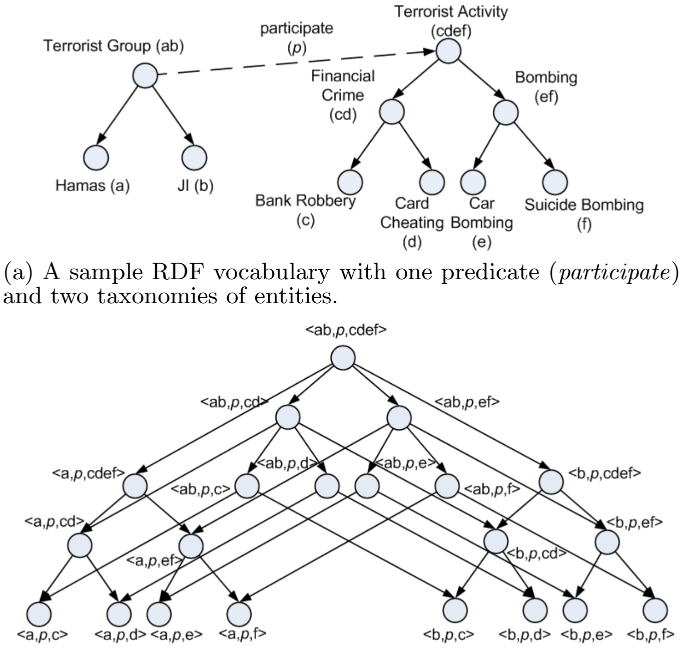
1. INTRODUCTION

Resource Description Framework (RDF) [4] is a specification proposed by the World Wide Web Consortium (W3C) for describing and interchanging semantic metadata on the Semantic Web [2]. Due to the continual popularity of the Semantic Web, in a foreseeable future, there will be a sizeable amount of RDF-based content available on the web, offering tremendous opportunities in discovering useful knowledge from large RDF databases. As one of the key data mining techniques in the area of Knowledge Discovery in Database (KDD), Association Rule Mining (ARM) [1] can play an important role for RDF based data mining in the Semantic Web era. With the use of the taxonomies of RDF entities (classes and instances) and RDF statements defined by RDF vocabularies, association rules can be extracted in a generalized form conveying knowledge in a compact manner. The discovered associations may have many applications, such as

- optimizing RDF storage and query processing,
- enhancing information source recommendation in Semantic Web, and
- association rule based (or frequent pattern based) classification and clustering of web resources.

2. APPROACH

RDF data sets are documents of RDF statements, each of which is a triplet in the form of <subject, predicate, object>. The unique characteristics of RDF data sets lie in the relatively large sizes of RDF documents and the complex taxonomic structures of the RDF statement hierarchies, wherein an RDF statement can be generalized in many ways, e.g. by generalizing its subject, object, predicate, or any combination of them (see Figure 1(a) and 1(b)). In ARM, *frequent pattern mining* (FPM) is the most time-consuming part. For RDF data sets, traditional generalized association rule mining algorithms that extract all frequent generalized patterns (RDF statement sets) do not work efficiently due to the fact that a large portion, if not most, of frequent generalized patterns are *over-generalized.* A frequent generalized pattern is said over-generalized if all RDF documents that contain the pattern also contain a same specialized pattern of it. For example, in Figure 1(c), the pattern X_2 is an over-generalization of X_1. In such a case, the pattern can always be inferred from its specialized pattern. Therefore, over-generalized patterns are redundant.



(a) A sample RDF vocabulary with one predicate (*participate*) and two taxonomies of entities.

(b) RDF statement hierarchy inferred from the RDF vocabulary in Figure 1(a).

Pattern $\boldsymbol{X_1}$ (subsumed by document 1, 2, and 3):
{<Terrorist Group, *participate*, Bank Robbery>, <Terrorist Group, *participate*, Car Bombing>}

Pattern $\boldsymbol{X_2}$ (subsumed by document 1, 2, and 3):
{<Terrorist Group, *participate*, Bank Robbery>, <Terrorist Group, *participate*, Bombing>}

Generalization Closure of $\boldsymbol{X_1}$:
{<Terrorist Group, *participate*, Bank Robbery>, <Terrorist Group, *participate*, Car Bombing>, <Terrorist Group, *participate*, Financial Crime>, <Terrorist Group, *participate*, Bombing>, <Terrorist Group, *participate*, Terrorist Activity>}

(c) Illustration of over-generalization and generalization closure.

Figure 1: Elements of RDF data sets.

WWW 2006, May 22–26, 2006, Edinburgh, Scotland.
ACM 1-59593-332-9/06/0005.

For accelerating the mining process, we employ the notion of *generalization closure* for full over-generalization reduction. A generalization closure of a pattern X, denoted as $\varphi_{gc}X$, is an RDF statement set containing all statements in X and all their generalized statements (see Figure 1(c)). A generalization closure is said to be *closed* if it does not have any superset of statements such that they are subsumed by the same set of RDF documents.

Generalization closures have the following key features:

1. The number of frequent closed generalization closures is usually substantially smaller than the number of all frequent generalized patterns.
2. All frequent generalized patterns can be derived from the set of all frequent closed generalization closures.
3. Given two RDF statement set X and Y, $\varphi_{gc}(X \cup Y) = \varphi_{gc}X \cup \varphi_{gc}Y$. This guarantees that we can gradually generate larger closures by merging smaller ones.

3. ALGORITHMS

We develop the GP-Close (Closed Generalized Pattern Mining) algorithm (Algorithm 1 and 2) that discovers the set of closed generalization closures instead of all frequent generalized patterns to minimize computation cost.

Algorithm 1 GP-Close

```
Input:
  RDF database: $\mathcal{D}$
  Generalized RDF statement lookup table: $GRT$
  Support Threshold: $minsup$
Output:
  The set of all closed frequent generalization closures: $\mathcal{C}$
$ce\_tree.root = \emptyset$ //initialize closure enumeration tree
$ce\_tree.root.supp = 1$
$ce\_tree.child\_gc\_list = \{\varphi_{gc}\{r\} | r$ is an RDF statement $\wedge supp(r) \geqslant minsup\}$ //Constructing child closure set of $ce\_tree.root$ from 1-frequent RDF statement set by looking up $GRT$.
Closure-Enumeration($ce\_tree$, $\mathcal{C} = \emptyset$)
return $\mathcal{C}$
```

Algorithm 2 Closure-Enumeration

```
Input:
  Closure enumeration tree: $ce\_tree$
  A set of discovered closed frequent generalization closures: $\mathcal{C}$
Output:
  The expanded set of closed frequent generalization closures: $\mathcal{C}$
if $\exists c \in \mathcal{C}$, with $c$ subsumes $ce\_tree.root$ then
    return $\mathcal{C}$ //prune the current sub enumeration tree
end if
$c$ = Closed-Closure($ce\_tree$) //Generate a closed generalization closure based on the current sub enumeration tree
$\mathcal{C} = \mathcal{C} \cup \{c\}$ //if c is not subsumed by a $c^* \in \mathcal{C}$
for each closure $gc_i \in ce\_tree.child\_gc\_list$ do
    $ce\_tree_i.root = gc_i$ //initialize a sub enumeration tree
    $ce\_tree_i.child\_gc\_list = \emptyset$
    for each $gc_j \in ce\_tree.child\_gc\_list$, with $i < j$ do
        $gc_{ij} = gc_i \cup gc_j$; $gc_{ij}.tidset = gc_i.tidset \cap gc_j.tidset$
        $gc_{ij}.supp = \frac{|gc_{ij}.tidset|}{|\mathcal{D}|}$
        $ce\_tree_i.child\_gc\_list = ce\_tree_i.child\_gc\_list \cup \{gc_{ij}\}$
    end for
    Closure-Enumeration($ce\_tree_i$, $\mathcal{C}$)
end for
return $\mathcal{C}$
```

4. RESULTS

Our experiments are conducted based on two real world RDF data sets, namely the foafPub data set provided by UMBC eBiquity Research Group[1] and the ICT-CB data set extracted from an online database of International Policy Institute for Counter-Terrorism (ICT)[2]. foafPub is a set of RDF files that describe peoples and their relationships with the use of the FOAF vocabulary[3]. The contents of the ICT-CB documents are descriptions of car bombing events. We also implemented the original generalized association rule mining algorithm, Cumulate [3], as a reference of performance evaluation and comparison.

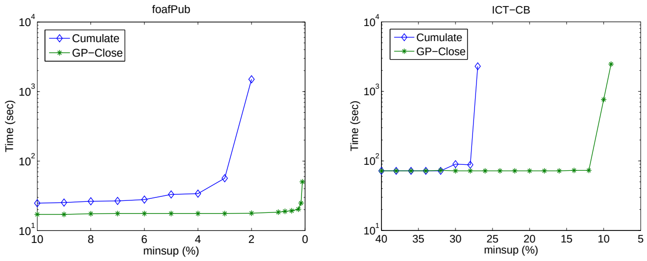


Figure 2: Execution time of GP-Close compared with Cumulate.

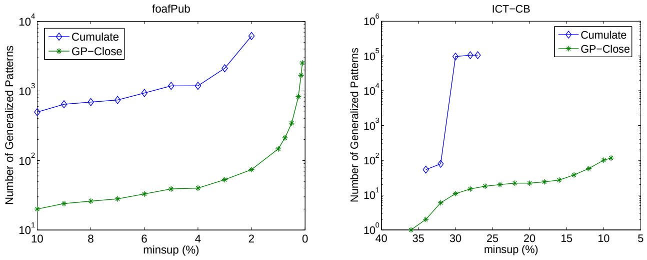


Figure 3: Number of patterns extracted by GP-Close compared with Cumulate.

Figure 2 shows the computation time of the two algorithms, Cumulate and GP-Close, with respect to the minimum support (*minsup*). We find that Cumulate can work properly only with high *minsup*. When the *minsup* is low, the GP-Close algorithm performs more than an order of magnitude faster than Cumulate.

The number of patterns discovered by the GP-Close and Cumulate are depicted in Figure 3. The number of closed generalization closures is almost one to two orders of magnitude smaller than the number of all frequent patterns discovered by Cumulate. Note that a *Log* scale is used in Figure 3. Therefore, the stable margin between the two curves actually implies an exponential growth in the difference between the numbers of all frequent patterns and the closed generalization closures.

5. REFERENCES

[1] R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In *SIGMOD '93*, pages 207–216, 1993.

[2] T. Berners-Lee, J. Hendler, and O. Lassila. Semantic web. *Scientific American*, 284(5):35–43, 2001.

[3] R. Srikant and R. Agrawal. Mining generalized association rules. In *VLDB '95*, pages 407–419, San Francisco, 1995.

[4] W3C. W3C RDF Specification.

[1] http://ebiquity.umbc.edu/resource/html/id/82/
[2] http://www.ict.org.il/
[3] http://xmlns.com/foaf/0.1/