Expert Systems with Applications 41 (2014) 1400–1410

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Misleading Generalized Itemset discovery

Luca Cagliero [*], Tania Cerquitelli, Paolo Garza, Luigi Grimaudo

Dipartimento di Automatica e Informatica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy

ARTICLE INFO

Keywords:
Generalized itemset mining
Data mining
Taxonomies
Mobile data analysis

ABSTRACT

Frequent generalized itemset mining is a data mining technique utilized to discover a high-level view of interesting knowledge hidden in the analyzed data. By exploiting a taxonomy, patterns are usually extracted at any level of abstraction. However, some misleading high-level patterns could be included in the mined set.

This paper proposes a novel generalized itemset type, namely the Misleading Generalized Itemset (MGI). Each MGI, denoted as $X \triangleright \mathcal{E}$, represents a frequent generalized itemset X and its set $\mathcal{E}$ of low-level frequent descendants for which the correlation type is in contrast to the one of X. To allow experts to analyze the misleading high-level data correlations separately and exploit such knowledge by making different decisions, MGIs are extracted only if the low-level descendant itemsets that represent contrasting correlations cover almost the same portion of data as the high-level (misleading) ancestor. An algorithm to mine MGIs at the top of traditional generalized itemsets is also proposed.

The experiments performed on both real and synthetic datasets demonstrate the effectiveness and efficiency of the proposed approach.

1. Introduction

Generalized itemset mining (Srikant & Agrawal, 1995) is an established data mining technique that focuses on discovering knowledge hidden in the analyzed data at different abstraction levels. By exploiting a taxonomy (i.e. a set of is-a hierarchies built over the analyzed data) the mining process entails discovering patterns, i.e. the frequent generalized itemsets, that (i) have a frequency of occurrence (support) in the analyzed data higher than or equal to a given threshold and (ii) can include items at any level of abstraction. Low-level itemsets represent rather specific and detailed data correlations for which the corresponding support is unlikely to exceed the given threshold. On the other hand, high-level (generalized) itemsets provide a high-level view of the underlying data correlations. Hence, they could represent, at a high granularity level, the knowledge that remains hidden at a lower abstraction level. The interestingness of an itemset is commonly measured in terms of the strength of the correlation between its items (Aggarwal & Yu, 1998; Brin, Motwani, & Silverstein, 1997; Savasere, Omiecinski, & Navathe, 1998). To evaluate itemset correlation, in this paper we exploit an established correlation measure, i.e. the Kulczynsky (Kulc) correlation measure (Wu, Chen, & Han, 2010). This measure has recently been adopted to perform high-level itemset correlation analysis (Barsky, Kim, Weninger, & Han, 2011). Itemset correlation values are usually clustered in three different correlation types. Specifically, if an itemset X occurs less than expected in the analyzed data (i.e. the item correlation value is between 0 and a given threshold *max_neg_cor*) then X is said to be *negatively correlated*; if it occurs more than expected (i.e. the item correlation value is above a given threshold *min_pos_cor*) then X shows a *positive correlation*, otherwise (i.e. whenever there is neither a positive nor a negative item correlation) X is said to be *not correlated*. Unfortunately, to support domain experts in making decisions not all of the mined high-level patterns can be trusted. Indeed, some misleading high-level itemsets could be included in the mining result. A generalized itemset X is, to some extent, misleading if (some of) the low-level X's descendants have a correlation type in contrast to those of X.

For example, let us consider the structured dataset that is reported in Table 1. Each record contains the record identifier (rid), the city, and the product description. The itemset mining process can be driven by the taxonomy in Fig. 1, which generalizes cities and products as the corresponding nations and product categories. Table 2 reports the set of frequent generalized itemsets that are mined by enforcing a support threshold min_sup = 1 and two correlation thresholds max_neg_cor = 0.65 and min_neg_cor = 0.8. The frequent generalized itemset X={(Product, Wearing), (City, Italy)} has a positive correlation type, whereas its frequent low-level descendant itemset Y={(Product, T-shirt), (City, Rome)} is negatively correlated (see Table 2). To estimate the extent to which X is misleading we evaluate the percentage of dataset records that

[*] Corresponding author. Tel.: +39 011 090 7084; fax: +39 011 090 7099.
E-mail addresses: luca.cagliero@polito.it (L. Cagliero), tania.cerquitelli@polito.it (T. Cerquitelli), paolo.garza@polito.it (P. Garza), luigi.grimaudo@polito.it (L. Grimaudo).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.08.039

Table 1
Example dataset $\mathcal{D}$.

Id	City	Product
1	Turin	T-shirt
2	Turin	T-shirt
3	Rome	T-shirt
4	Paris	Jacket
5	Paris	Jacket
6	Cannes	Book
7	Turin	T-shirt

Table 2
MGI mined from $\mathcal{D}$. min_sup = 1, max_neg_cor = 0.65, min_pos_cor = 0.80, and max_NOD = 100%.

Frequent generalized itemset (level $\geqslant 2$) [correlation type (Kulc value)]	Frequent descendant [correlation type (Kulc value)]	Not overlapping degree (%)
{(City, Italy)} [positive (1)]	{(City, Turin)} [positive (1)] {(City, Rome)} [positive (1)]	–
{(City, France)} [positive (1)]	{(City, Paris)} [positive (1)] {(City, Cannes)} [positive (1)]	–
{(Product, Wearing)} [positive (1)]	{(Product, T-shirt)} [positive (1)] {(Product, Jacket)} [positive (1)]	–
{(Product, Education)} [positive (1)]	{(Product, Book)} [positive (1)]	–
{(Product, Wearing), (City, Italy)} **[positive (5/6 = 0.83)]**	{(Product, T-shirt), (City, Turin)} **[positive (7/8 = 0.88)]** {(Product, T-shirt), (City, Rome)} **[negative (5/8 = 0.63)]**	**75**
{(Product, Wearing), (City, France)} **[negative (1/2 = 0.50)]**	{(Product, Jacket), (City, Paris)} **[positive (1)]**	**0**
{(Product, Education), (City, France)} **[negative (2/3 = 0.66)]**	{(Product, Book), (City, Cannes)} **[positive (1)]**	**0**

are covered by both X and any of its contrasting low-level correlations. For example, the record with rid 3 is covered by both X and Y. In other words, 25% of the records that are covered by {(Product, Wearing), (City, Italy)} are in common with those covered by {(Product, T-shirt), (City, Rome)}.

In this paper we propose: (i) a novel generalized itemset type, namely the Misleading Generalized Itemset (MGI); (ii) a MGI quality measure called Not Overlapping Degree (NOD) which indicates the extent to which the high-level pattern is misleading compared to its low-level descendants; and (iii) an approach to discovering a worthwhile subset of MGIs with NOD less than or equal to a maximum threshold *max_NOD*. Specifically, each MGI, hereafter denoted as $X \triangleright \mathcal{E}$, represents a frequent generalized itemset X and its set $\mathcal{E}$ of low-level frequent descendants for which the correlation type is in contrast to those of X. Experts need to analyze the misleading high-level data correlations separately from the traditional generalized itemsets and exploit such knowledge by making different decisions. To make this analysis possible, MGIs are extracted only if the low-level descendant itemsets that represent contrasting correlations cover almost the same portion of data as the high-level (misleading) ancestor X, i.e. only if X represents a "clearly misleading" pattern. To do so, a maximum NOD constraint is enforced during the MGI mining process. Hence, unlike previous approaches (e.g. Barsky et al., 2011; Brin et al., 1997), we evaluate the degree of overlapping between the sets of records that are covered by a generalized itemset and its low-level (descendant) contrasting correlations. An algorithm to mine MGIs at the top of traditional generalized itemsets is also proposed.

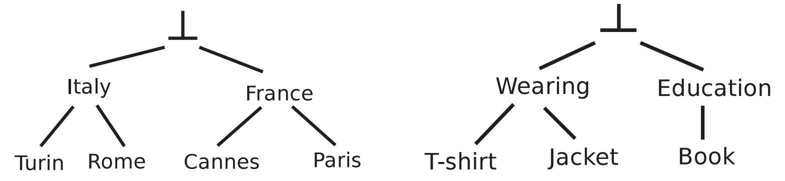
The effectiveness of the proposed approach and the usability of the discovered patterns for supporting domain expert decisions are demonstrated by experiments performed on real-life data coming from two mobile applications and the UCI data repository (Blake & Merz, 2012). Furthermore, the scalability of the algorithm has also been evaluated on synthetic datasets.

This paper is organized as follows: Section 2 introduces preliminary definitions; Section 3 formally states the MGI mining problem; Section 4 describes the MGI MINER algorithm; Section 5 discusses the performed experiments; Section 6 presents previous works and Section 7 draws conclusions and discusses some possible future developments of this work.

2. Preliminary definitions and notations

This paper addresses the problem of generalized itemset mining from structured data that are supplied with taxonomies. A structured dataset is a set of records. Each record is a set of items, which are defined as pairs (attribute_name, value). While attribute_name is the description of a data feature, value represents the associated information and belongs to the corresponding attribute domain. Since continuous attribute values are unsuitable for use in itemset mining, continuous values are discretized by a traditional preprocessing step (Tan, Steinbach, & Kumar, 2005). For instance, Table 1 reports an example of structured dataset D that is composed of 3 attributes: the record identifier (rid), the city, and the product description.

A taxonomy is a set of is-a hierarchies built over the data attribute items. It consists of a set of aggregation trees, one or more for each dataset attribute, in which the items that belong to the same attribute domain are aggregated in higher level concepts. For example, let us consider the taxonomy that is reported in Fig. 1. It includes two aggregation trees, one for each attribute in $\mathcal{D}$. By construction, we disregard the rid attribute for the subsequent analysis. For each aggregation tree the leaf nodes are labeled with values belonging to the corresponding attribute domain, whereas each non-leaf node aggregates (a subset of) lower level nodes



(a) Aggregation tree $AT_{location}$ defined on the Location attribute

(b) Aggregation tree $AT_{product}$ defined on the Product attribute

Fig. 1. Example taxonomy built on $\mathcal{D}$'s attributes.