A Similarity Function for Multi-Level and Multi-Dimensional Itemsets

Matteo Francia, Matteo Golfarelli, and Stefano Rizzi

DISI, University of Bologna, Italy

Abstract. The key objective of frequent itemsets (FIs) mining is uncovering relevant patterns from a transactional dataset. In particular we are interested in multi-dimensional and multi-level transactions, i.e., ones that include different points of view about the same event and are described at different levels of detail. In the context of a work aimed at devising original techniques for summarizing and visualizing this kind of itemsets, in this paper we extend the definition of itemset containment to the multi-dimensional and multi-level scenario, and we propose a new similarity function for itemsets, enabling a more effective grouping. The most innovative aspect of our similarity function is that it takes into account both the extensional and intensional natures of itemsets.

Keywords: Frequent itemset mining, Itemset summaries

1 Introduction

The key objective of frequent itemsets (FIs) mining is uncovering relevant patterns from a transactional dataset [2]. Since its initial formulation on transactions of uniform and flat items, where a transaction corresponds to a set of products bought together by a customer, FIs mining has been applied to different types of data. In particular, in this work we consider *multi-dimensional* and *multi-level* data [8]. A multi-dimensional transaction represents an event from different points of views, which we call *features*. In a multi-level transaction feature values are described using hierarchies with different levels of detail. A typical application is that of user profiling: each transaction describes a user by means of several features related for instance to where she lives, where she works, how much she earns; each feature values can be described at different, hierarchically-organized levels (e.g., she lives close to Macy's, which is in the Garment district, which is part of Manhattan). In this context, a FI describes a profile of a group of people sharing the same features/behavior.

The exponential nature of FIs [2] makes it difficult for data scientists and domain experts to visualize and explore their information content. Increasing the frequency threshold (i.e., the minimum itemset support) just decreases the number of FIs possibly leading to missing useful information; so, it is recognized that more effective approach is that of providing FI *summaries* to assist decision

SEBD 2018, June 24-27, 2018, Castellaneta Marina, Italy.

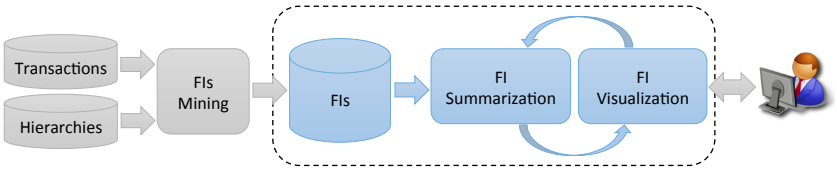

Fig. 1. A functional architecture of a framework to summarize and visualize FIs

makers in getting insights over data [14]. Summarization differs from FIs mining since the latter searches, within a set of transactions, those itemset that are frequently present disregarding redundancy. Conversely, summarization is an optimization problem that addresses the extraction of the minimum number of FIs that represent an entire population while maximizing the overall diversity of the representatives. The two techniques are complementary to enhance and simplify FIs analysis: the adoption of summarization on top of FIs mining makes the choice of a frequency threshold less critical and enables the discovery of both specific and general patterns. Though several FI summarization approaches have been proposed in the literature (e.g., [1,12,5]), they do not consider the multi-level and multi-dimensional natures of FIs. A third approach to make FI analysis more effective is the use of advanced visualizations. Interactive visual interfaces and visual data mining approaches can unveil hidden information, simplify the process of understanding, and allow users to focus their attention on what is important. Although some visual representations for FIs have been proposed in the literature [13,9,4], to the best of our knowledge no approaches for visualizing FIs summaries have been proposed so far.

To fill this gap, we are currently working on a framework addressing the summarization and visualization of multi-level and multi-dimensional FIs. As shown in Figure 1, our approach is independent of the algorithm applied for generating the FIs taken in input [11,3]. The summarization and the visualization components work jointly to give users the relevant information; the user can iteratively create and visualize new summaries that better meet her needs by tuning a set of parameters. In the context of this framework, here we extend the definitions of FIs and itemset containment to the multi-dimensional and multi-level scenario, and we propose a new similarity function for FIs that enables more effective groupings. The most innovative aspect of our similarity function is that it takes into account both the extensional and intensional natures of FIs. The intensional nature is considered in *feature-based similarity*: the higher the number of features (i.e., semantics) shared by two FIs, the higher their similarity; the extensional nature is considered in *support-based similarity*: the higher the percentage of transactions supporting both FIs, the higher their similarity. Adopting this two-faceted similarity function, agglomerative clustering algorithms [7] can then be leveraged to summarize FIs.

The paper is organized as follows. After providing the formal definitions of multi-dimensional and multi-level FIs in Section 2, in Section 3 we propose our similarity function. Finally, in Section 4 we discuss the research perspectives.

2 Itemsets

The itemsets we consider are multi-level, which implies the presence of a hierarchy of concepts. The type of hierarchies we consider are those defined in classic multi-dimensional modeling [6].

Definition 1 (Hierarchy). *A* hierarchy H *is defined by (i) a set* L_H *of categorical* levels*, (ii) a* domain $Dom(l)$ *including a set of* values *for each level* $l \in L_H$ *(all domains are disjoint), (iii) a* roll-up *partial order* $\succeq_H$ *of* L_H*, and (iv) a* part-of *partial order* $\geq_H$ *of* $\bigcup_{l \in L_H} Dom(l)$*. Exactly one level* $dim(H) \in L_H$*, called* dimension*, is such that* $dim(H) \succeq_H l$ *for each other* $l \in L_H$*. The part-of partial order is such that, for each couple of levels* l *and* l' *such that* $l \succeq_H l'$*, for each value* $v \in Dom(l)$ *there is exactly one value* $v' \in l'$ *such that* $v \geq_H v'$*.*

The itemsets we consider are also multi-dimensional, i.e., they refer to different features (e.g., worksIn) each related to a specific hierarchy (e.g., Location). A feature defines the semantics carried by an item at a specific hierarchical level. This can be formalized as follows:

Definition 2 (Domain Schema). *A* domain schema *is a triple* $\mathcal{D} = (\mathcal{H}, \mathcal{F}, \mu)$ *where: (1)* $\mathcal{H}$ *is a set of hierarchies; (2)* $\mathcal{F}$ *is a set of* features*; and (3)* μ *is a function mapping each feature onto one hierarchy.*

Example 1. As a working example we will use the Profiling domain schema, which describes the customers who regularly visit a mall and features the two hierarchies depicted in Figure 2. The first one is rooted in the Location dimension and has two branches: the first one describes locations from the geographical point of view (with reference to New York City), the second one based on their features. In the roll-up partial order we have, for instance, $\mathsf{Neighborhood} \succeq_{\mathsf{Location}} \mathsf{Borough}$; in the part-of partial order, we have $\mathrm{Harlem} \geq_{\mathsf{Location}} \mathrm{Manhattan}$. The second hierarchy describes incomes in terms of their ranges. The features of Profiling are worksIn, frequents, and earns; specifically,

$$\mu(\mathsf{worksIn}) = \mu(\mathsf{frequents}) = \mathsf{Location},\ \ \mu(\mathsf{earns}) = \mathsf{Income}$$

Itemsets are non-redundant sets of items, i.e., two items in an itemset cannot be defined on values related in the part-of partial order (e.g., GreenwichVillage and Manhattan). Finally, transactions are itemsets whose items are all defined on dimension values (e.g., Macy's).

Definition 3 (Itemset and Transaction). *Given domain schema* $\mathcal{D} = (\mathcal{H}, \mathcal{F}, \mu)$*, an* item *of* $\mathcal{D}$ *is a couple* $i = (f, v)$ *where* $f \in \mathcal{F}$*,* $v \in Dom(l)$*, and* l *is a level of hierarchy* $\mu(f)$*. An* itemset I *of* $\mathcal{D}$ *is a set of distinct items of* $\mathcal{D}$ *where, for each* $i, i' \in I$*, with* $i = (f, v)$ *and* $i' = (f, v')$*, it is* $v \not\geq_{\mu(f)} v'$ *and* $v' \not\geq_{\mu(f)} v$*. A* transaction *is an itemset only including items defined over dimensions of* $\mathcal{H}$*.*

Example 2. Examples of itemset I and transaction T of Profiling are

$$I = \{(\mathsf{worksIn}, \mathrm{Harlem}), (\mathsf{frequents}, \mathrm{Museum}), (\mathsf{earns}, \mathrm{High})\}$$
$$T = \{(\mathsf{worksIn}, \mathrm{CityCollege}), (\mathsf{frequents}, \mathrm{WhitneyMuseum}), (\mathsf{earns}, 35\mathrm{to}60)\}$$

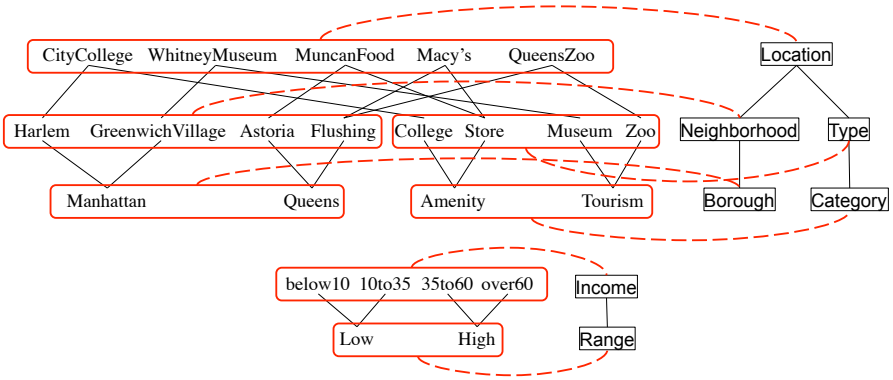

Fig. 2. Hierarchies (right) and values (left) for the Profiling domain schema; in red, the level-domain mappings

Working with multi-dimensional and multi-level items requires to extend the classic definition of *containment* between itemsets, which generalizes set containment taking hierarchies into account.

Definition 4 (Itemset Containment). *Given two itemsets I and I', we say that I is* contained *in I' (denoted $I \sqsubseteq I'$) iff for each item $i \in I$, $i = (f, v)$, there is an item $i' \in I'$, $i' = (f, v')$ such that $v' \geq_{\mu(f)} v$.*

Let $\mathcal{I}$ denote the set of all items of a schema domain. It can be easily verified that the containment relationship is reflexive, antisymmetric, and transitive, and that for each pair of itemsets in $\mathcal{I}$ there are a least upper bound and a greatest lower bound; so $\sqsubseteq$ induces a lattice on $2^{\mathcal{I}}$. The top element of the lattice is the empty itemset, the bottom element is $\mathcal{I}$. Given two itemsets I and I', we denote with $lub(I, I')$ and $glb(I, I')$ their least upper bound and greatest lower bound.

Example 3. Figure 3 shows a small portion of the containment lattice for Profiling; for simplicity we restrict to features frequents and earns and denote items by their value only. For instance, for frequents it is $\{\text{Amenity}\} \sqsubseteq \{\text{College}, \text{Store}\}$ and $\{\text{Amenity}\} \sqsubseteq \{\text{College}\}$. Besides, it is

$$lub(\{\text{CityCollege}\}, \{\text{College}, \text{Store}\}) = \{\text{CityCollege}, \text{Store}\}$$
$$glb(\{\text{CityCollege}\}, \{\text{College}, \text{Store}\}) = \{\text{College}\}$$

Transaction T is said to *support* itemset I iff $I \sqsubseteq T$. With reference to Example 2, T supports I. Given a set of transactions $\mathcal{T}$, the set of transactions that support I is denoted by $\mathcal{T}_I \subseteq \mathcal{T}$. This allows us to introduce a relevant numerical property of itemsets, namely, their *support*.

Definition 5 (Itemset Support). *Given itemset I, its support $sup(I)$ within a set of transactions $\mathcal{T}$ is defined as*

$$sup(I) = \frac{|\mathcal{T}_I|}{|\mathcal{T}|}$$

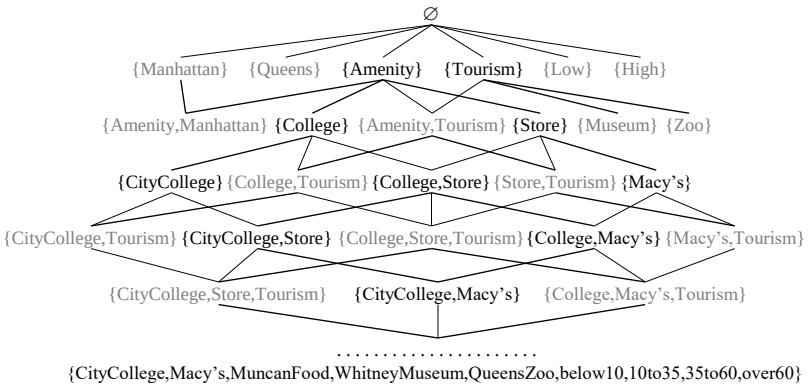

Fig. 3. A portion of the containment for Profiling, including containment arcs; itemsets in gray are not completely expanded

Itemset I is said to be *frequent* if it is greater or equal to a given threshold. Note that, since the containment relationship induces a lattice on the set $2^{\mathcal{I}}$ of all possible itemsets, it also induces a partial order over the set $\mathcal{F} \subseteq 2^{\mathcal{I}}$ of FIs. Thus, from now on we will say that $\mathcal{F}$ is a POS (Partially Ordered Set).

3 Itemset Similarity

We argue that itemset similarity is a two-faceted concept: (1) according to *feature-based* similarity, the higher the number of features (i.e., semantics) shared by two FIs, the higher their similarity; and (2) feature-based similarity is useless if two FIs include two distinct groups of transactions, thus, it is complemented by *support-based* similarity in which the higher the percentage of transactions supporting both FIs, the higher their similarity. These two aspects of similarity are not necessarily correlated; for example, support-based similarity can be low even if feature-based similarity is high when non-shared features are rare and supported by a small fraction of transactions.

In a multi-level and multi-dimensional domain, computing feature-based similarity is not just a matter of finding the subset of common items between two FIs, but it is also related to the informative value they carry in terms of level of detail. Intuitively, we consider an FI to be more *relevant* than another if it includes a larger number of distinct features; in turn, the relevance of a feature increases with the level of detail at which it is expressed.

Definition 6 (Itemset Relevance). *Given itemset I, its* relevance *is defined as*

$$rel(I) = \sum_{f \in Feat(I)} \left(rel(f) + \sum_{l \in Lev_f(I)} rel(l) \right)$$

where $Feat(I)$ is the set of distinct features of the items in I, $Lev_f(I)$ is the set of levels of the values coupled with feature f in the items of I, $rel(f)$ is the relevance of f, and $rel(l)$ is the relevance of level l. Conventionally, $rel(\varnothing) = 0$.

We finally introduce the similarity between two FIs as a linear combination of a support-based and a feature-based similarity.

Definition 7 (Itemset Similarity). *Given a set of transactions $\mathcal{T}$, a POS of FIs $\mathcal{F}$, two FIs I and I' supported by $\mathcal{T}$, and a coefficient $\lambda \in [0..1]$, the* similarity *of I and I' is defined as*

$$sim(I, I') = \lambda sim_{sup}(I, I') + (1 - \lambda) sim_{rel}(I, I')$$

where

$$sim_{sup}(I, I') = \frac{sup(glb(I, I'))}{sup(I) + sup(I') - sup(glb(I, I'))}$$

$$sim_{rel}(I, I') = \begin{cases} \frac{rel(lub(I,I'))}{rel(glb(I,I'))}, \text{ if } lub(I, I'), glb(I, I') \in \mathcal{F} \\ 0, \text{ otherwise} \end{cases}$$

Both sim_{sup} and sim_{rel} range in $[0..1]$ and can be intuitively explained as follows: sim_{sup} is the ratio between the number of transactions supporting both FIs I and I' and the number of transactions supporting either I or I'; sim_{rel} is the ratio between the relevance of the features common to I and I' and the relevance of the union of the features of I and I'. Clearly, since the *lub* and *glb* operators are commutative, it is always $sim(I, I') = sim(I', I)$.

Example 4. With reference to the hierarchies defined in Figure 2, we assume that (i) all features are equally relevant ($rel(f) = 1$ for all f), and (ii) relevance increases by 0.1 for each level of detail. Given FIs $I = \{(\mathsf{frequents}, \text{College}), (\mathsf{worksIn}, \text{Store})\}$ and $I' = \{(\mathsf{frequents}, \text{College}), (\mathsf{frequents}, \text{Store})\}$, it is

$$L = lub(I, I') = \{(\mathsf{frequents}, \text{College}), (\mathsf{frequents}, \text{Store}), (\mathsf{worksIn}, \text{Store})\}$$
$$G = glb(I, I') = \{(\mathsf{frequents}, \text{College})\}$$

Assuming for instance that $sup(I) = 0.3, sup(I') = 0.4, sup(L) = 0.2, sup(G) = 0.5$, and $\lambda = 0.5$, then $sim(I, I') = 0.44$.

4 Discussion

In this paper we have proposed an original similarity measure for multi-dimensional and multi-level FIs, to be used for enabling the creation of concise and valuable summaries of sets of FIs. Though for space reasons we cannot fully detail how summaries are defined and visualized in our approach, in this section we give an informal explanation.

First of all, since our goal is to support interactive exploration and navigation of FIs, we organize summaries in a hierarchical fashion so that they can be

analyzed at different levels of detail. So we build a hierarchical clustering of the set of all FIs using an agglomerative algorithm; any (complete and disjoint) "cut" of the resulting dendrogram is a summary.

In a summary each cluster is represented by a single FI; specifically, the representative of cluster c is the most specific FI in c, i.e., such that $I \sqsubseteq rep(c), \forall I \in c$. With reference to this, we note that the strategy commonly used in the literature picks as a cluster representative its most general FI [5]; however, this strategy often lacks in properly characterizing clusters since it easily yields very low relevance as the cluster cohesion decreases, which may entail one or more features appearing in some of the cluster FIs to be missing from the representative. Conversely, when using our strategy, all the features appearing in at least one FI of the cluster are included in the representative.

As to the agglomerative clustering algorithm we adopt, it is basically a greedy algorithm that progressively merges couples of clusters starting from singletons and until one single cluster is obtained. Remarkably, the POS of FIs induced by our definition of itemset containment allows the search space to be significantly pruned. Indeed, our preliminary tests show that the POS obtained when a feature is described using a linear hierarchy of n levels is several orders of magnitude smaller than the one we would get if those n levels were flat. Another relevant improvement we get in terms of computational complexity depends on an interesting property of our similarity function. It can be proved that, in domains where the relevance of the levels of a hierarchy increases with the level of detail (i.e., $rel(l) \geq rel(l')$ if $l \succeq_H l'$), similarity is antimonotonic along the itemset containment relationship. As a consequence, at each step of our agglomerative algorithm we can just estimate the similarity between FIs that are *directly* contained into one another, thus avoiding a large number of useless computations.

Finally, to visualize summaries we adopt treemaps, a popular method for visualizing large hierarchical data sets by mapping hierarchical concepts into 2D areas [10]. Figure 4 shows a treemap in which the visualization area is partitioned into nested rectangles, each corresponding to a cluster of FIs whose area is proportional to the cluster cardinality; colors code both the predominant feature (i.e., the one with the highest relevance within the cluster FIs) of the cluster representative (hue) and its support (saturation). So, for instance, the top right pink rectangle describes a cluster that (i) includes 97 FIs with support ranging from 0.14 to 0.58 and relevance ranging from 1.00 to 3.60; (ii) has 10 child clusters in the dendrogram; and (iii) has a representative (namely, {(earns, avg30), (livesIn, close), (livesIn, collina_BO), (worksIn, Bologna), (worksIn, close)}) whose predominant feature is livesIn. On this visualization the user can then apply classical OLAP operators (roll-up, drill-down, slice-and-dice) to navigate the dendrogram so as to flexibly explore the set of FIs at different abstraction levels and focusing on the more relevant FIs.

References

1. Afrati, F.N., Gionis, A., Mannila, H.: Approximating a collection of frequent sets. In: Proc. SIGKDD. pp. 12–19. Seattle, USA (2004)

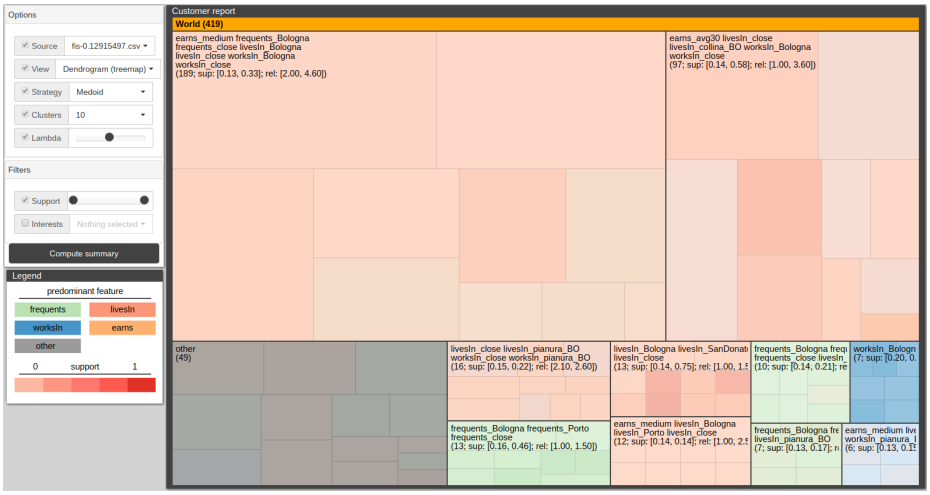

Fig. 4. Visualization of summaries

2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proc. VLDB. pp. 487–499. Santiago, Chile (1994)
3. Baralis, E., Cagliero, L., Cerquitelli, T., D'Elia, V., Garza, P.: Support driven opportunistic aggregation for generalized itemset extraction. In: Proc. IS. pp. 102–107. London, UK (2010)
4. Bothorel, G., Serrurier, M., Hurter, C.: Visualization of frequent itemsets with nested circular layout and bundling algorithm. In: Proc. ISVC. pp. 396–405. Rethymnon, Greece (2013)
5. Chandola, V., Kumar, V.: Summarization — compressing data into an informative representation. Knowl. Inf. Syst. 12(3), 355–378 (2007)
6. Golfarelli, M., Rizzi, S.: Data warehouse design: Modern principles and methodologies. McGraw-Hill (2009)
7. Guha, S., Rastogi, R., Shim, K.: CURE: an efficient clustering algorithm for large databases. In: Proc. SIGMOD. pp. 73–84. Washington, USA. (1998)
8. Han, J., Fu, Y.: Mining multiple-level association rules in large databases. IEEE Trans. Knowl. Data Eng. 11(5), 798–804 (1999)
9. Leung, C.K., Carmichael, C.L.: FpViz: a visualizer for frequent pattern mining. In: Proc. SIGKDD. pp. 30–39. Paris, France (2009)
10. Shneiderman, B.: Tree visualization with tree-maps: 2-d space-filling approach. ACM Trans. Graph. 11(1), 92–99 (1992)
11. Srikant, R., Agrawal, R.: Mining generalized association rules. In: Proc. VLDB. pp. 407–419. Zurich, Switzerland (1995)
12. Wang, J., Karypis, G.: On efficiently summarizing categorical databases. Knowl. Inf. Syst. 9(1), 19–37 (2006)
13. Yan, X., Cheng, H., Han, J., Xin, D.: Summarizing itemset patterns: a profile-based approach. In: Proc. SIGKDD. pp. 314–323. Chicago, USA (2005)
14. Zhang, S., Jin, Z., Lu, J.: Summary queries for frequent itemsets mining. Journal of Systems and Software 83(3), 405–411 (2010)