

Using Fuzzy Ontologies to Extend Semantically Similar Data Mining

Eduardo L. G. Escovar, Cristiane A. Yaguinuma, Mauro Biajiz

Department of Computer Science – Federal University of São Carlos (UFSCar)
P.O. Box 676 – 13565-905 – São Carlos – SP – Brazil

{escovar, cristiane_yaguinuma, mauro}@dc.ufscar.br

Abstract. Association rule mining approaches traditionally generate rules based only on database contents, and focus on exact matches between items in transactions. In many applications, however, the utilization of some background knowledge, such as ontologies, can enhance the discovery process and generate semantically richer rules. Besides, fuzzy logic concepts can be applied on ontologies to quantify semantic similarity relations among data. In this context, we extended SSDM (Semantically Similar Data Miner) algorithm in order to obtain from a fuzzy ontology the semantic relations between items. As a consequence, the generated rules can be more understandable, improving the utility of the knowledge supplied by them.

1. Introduction

Among all data mining tasks, association rule mining is probably the most used one due to its utility in many applications. Given a set of transactions, where each transaction is a set of items, an association rule is an expression $X \Rightarrow Y$, where X and Y are sets of items (or itemsets). The meaning of such a rule is that transactions which contain items in X tend to also contain items in Y . The support of the rule $X \Rightarrow Y$ is the percentage of transactions that contain both X and Y . The confidence of the rule $X \Rightarrow Y$ is the percentage of transactions containing X that also contain Y . Traditionally, the problem of mining association rules is to find rules having minimum support and confidence.

Approaches in this area are generally motivated by finding new ways of dealing with different attribute types or increasing computational performance. However, a crescent number of approaches have been concerned about semantics on mined data, aiming to improve the quality of discovered knowledge. This is the case of SSDM (Semantically Similar Data Miner) [Escovar et al. 2005], which considers data semantics to reveal more understandable association rules. In order to generate these richer rules, SSDM uses fuzzy logic concepts to define the semantic similarity between items. Such semantic information should be previously specified by an expert user.

Similarly, Semantic Web also demands the representation of semantic information defined by knowledge experts. In this scenario, ontologies have been adopted as a standard formalism to represent semantics. By definition, ontologies express the structure and the meaning of concepts and relationships of a domain. Moreover, they are usually constructed by domain experts, resulting in a consensual and shared knowledge. Since ontology semantics is widely reused and accepted by many communities and applications, it becomes natural to use it on association rule mining approaches which consider semantic information.

However, the conceptual formalism supported by typical ontologies may not be sufficient to represent uncertain information that is commonly found in many application domains [Quan et al. 2004]. For example, a concept that describes race or ethnicity like *Brown*, can be more semantically related to concept *Black* than to concept *White*, so it is inappropriate to treat all relationships equally as some of them may be more significant than others. To deal with this kind of problem, one alternative is to incorporate fuzzy logic concepts into ontologies so that it can be possible to handle uncertainty on data.

Given this context, our work extends the SSDM algorithm in order to use ontologies as background knowledge to represent semantics over the mined data. Consequently, SSDM can reuse consensual and shared knowledge, easing the process of acquiring semantic information. As SSDM considers uncertainty on data, fuzzy logic concepts are incorporated into the used ontologies, resulting in fuzzy ontologies that can capture richer semantics than crisp ontologies. These fuzzy ontologies include similarity degree values between concepts, which are processed by SSDM to generate more understandable association rules that reflect the semantic similarity among data.

The rest of this paper is organized as follows. Section 2 discusses related work on association rule mining approaches that use ontologies and/or fuzzy concepts. Section 3 describes how we extended SSDM to handle the fuzzy ontology used in our work. Performed experiments are discussed in Section 4. Finally, Section 5 presents our conclusions.

2. Related Work

Ontologies have been applied as background knowledge in data mining to enhance the discovery process. Basically, they can play several roles in the Knowledge Discovery in Databases (KDD) process: they can be applied for understanding the application problem, preparing and mining the data, generating high level rules and, last but not least, for analyzing interestingness on discovered rules or patterns.

The work described in [Chen et al. 2003], which uses an ontology to improve support in rule mining, is an example of approach that consider semantic information during the preprocessing step. In this work, data is raised to more generalized concepts according to the ontology, and then the mining process is performed by a conventional association rule mining algorithm like Apriori [Agrawal and Srikant 1994]. Authors argue that previous data generalization makes possible to consider subcategories in support calculation, generating rules with higher support. Furthermore, obtained rules can be easier to interpret, since they contain high level concepts that represent richer information than specific terms in database.

Likewise, relevant work has focused on the post-processing step. In [Hou et al. 2005], for example, domain knowledge is used to generalize low level rules discovered by usual rule mining algorithms, in order to obtain fewer and clearer high level rules. Authors use ontologies to generalize the concepts in rules after applying the core data-mining algorithm, and then they apply the data mining algorithm again to discover the high level in the abstract rules. Another example is described in [Pohle 2003], where ontologies are employed to determine rule interestingness. This is done by verifying whether discovered rules confirm, contradict or reveal new information when compared to the knowledge available in the ontology. Furthermore, the author also proposes feed-

back mechanisms to update domain knowledge from generated rules, because new and interesting insights can be discovered from the results of the mining process.

Other approaches, like ExCIS [Brisson et al. 2005], use domain knowledge in both pre and post-processing steps. In this work, the preprocessing step uses an ontology to guide the construction of specific datasets for particular mining tasks. The next step is the application of a standard mining algorithm which extracts patterns from these datasets. Finally, in the post-processing step, mined rules may be interpreted and/or filtered, as their terms are generalized according to an ontology. Therefore, semantic information used in ExCIS supports dataset preparation and allows reducing the volume of extracted patterns.

In summary, referred work has used ontologies mainly as concept hierarchies or taxonomies, focusing on generalization relationships between concepts. Such background knowledge was used in order to obtain a reduced number of rules that are more interesting and understandable to the end user. Although domain knowledge has an important role to improve mining results, one bottleneck faced by aforementioned approaches is that the conceptual formalism supported by typical ontology may not be sufficient to represent uncertain information found in many applications [Quan et al. 2004]. This is because general ontologies contain crisp inter-concept relations and can not quantify the strength of a relation. According to Wallace and Avrithis [Wallace and Avrithis 2004], relations among real life entities are always a matter of degree, and are, therefore, best modeled using fuzzy relations. For this reason, it is suitable to incorporate fuzzy logics into domain knowledge, in order to handle data uncertainty. Thus, some association rule mining approaches have been using fuzzy concepts in taxonomies or concept hierarchies so that membership degree can be considered when computing support and confidence of association rules.

Chen, Wei and Kerre's work [Chen et al. 2000] focuses on the matter of mining generalized association rules with fuzzy taxonomic structures. While conventional taxonomies have a child belonging to its ancestor with degree 1, on fuzzy taxonomies a child can belong to its ancestor with degree μ ($0 \leq \mu \leq 1$). The authors extended the algorithm proposed by Srikant and Agrawal [Srikant and Agrawal 1995] so that the computation of support and confidence could be applied in a fuzzy context. After that, Chen and Wei have developed another work [Chen and Wei 2002], where linguistic hedges were also incorporated in mining fuzzy rules to express more meaningful knowledge.

Another work that also considers fuzzy logic, taxonomies and data mining is described by Hong, Lin and Wang [Hong et al. 2003]. The algorithm proposed by them integrates fuzzy set concepts and generalized data mining to find cross-level interesting rules from quantitative data. In order to accomplish that, item quantities are transformed into fuzzy sets and fuzzy rules are generated by modifying Srikant and Agrawal's method [Srikant and Agrawal 1995] to manage hierarchical fuzzy items. Association rules are said to be cross-level because quantitative items may belong to any level of the given taxonomy. Since mined rules are expressed in fuzzy linguistic terms belonging to different semantic levels, information can be more natural and easily understandable by users.

3. Fuzzy Domain Knowledge and Semantically Similar Data Mining

The related work presented in Section 2 introduced different ways of using domain knowledge to obtain association rules, often achieving promising results. In particular, ap-

proaches which involved fuzzy sets concepts and taxonomies obtained a better way of modeling relations among entities. However, these approaches can be inappropriate in some situations, because generalization of terms may induce wrong interpretation of rules, even when using fuzzy domain knowledge. In this section, we show our proposal to deal with this problem and present how the SSDM algorithm [Escovar et al. 2005] was modified in order to handle fuzzy ontologies.

3.1. Problems Related to Generalized Association Rule Mining

In general, current work regarding fuzzy ontologies or taxonomies uses fuzzy membership degree in *is-a* relationships between concepts. Some works from Information Retrieval [Widyantoro and Yen 2001] [Parry 2004] and from Ontology Generation [Quan et al. 2004] consider fuzzy ontologies according to this approach, which makes possible to quantify how concepts are related to their ancestors. There are some generalized association rule mining approaches, like the work proposed by Chen, Wei and Kerre [Chen et al. 2000] and Chen and Wei [Chen and Wei 2002], that also incorporate fuzzy membership degrees into taxonomies. An example of such fuzzy taxonomy is shown in Figure 1, which is based on the fuzzy taxonomy presented in [Chen et al. 2000]. It illustrates a concept hierarchy of food items and membership degrees between concepts and their ancestors.

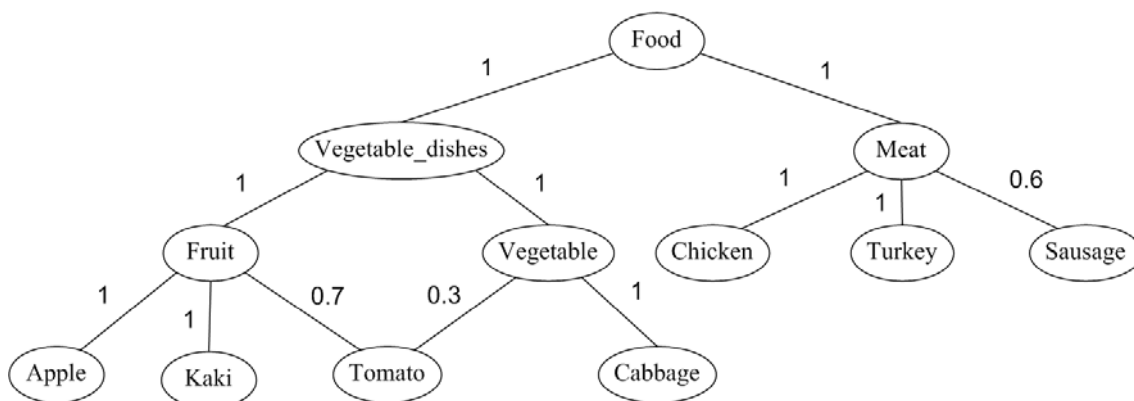


Figure 1. Fuzzy taxonomy of food items.

According to Chen, Wei and Kerre [Chen et al. 2000], fuzzy taxonomic structures can reflect partial belonging of one item to another, and therefore make possible to capture richer semantics than traditional domain knowledge representations. In the example presented in Figure 1, *Sausage* partially belongs to *Meat*, and *Tomato* may be regarded as being both *Fruit* and *Vegetable*, but to different degrees. Authors claim that such partial relationships can not be expressed by a crisp concept hierarchy. Fuzzy taxonomies use membership degrees in *is-a* relationships, representing that a sub-item belongs to its super-item with a certain degree μ ($0 \leq \mu \leq 1$). Another important point to consider, as stated by Shekar and Natarajan [Shekar and Natarajan 2004], is that there are no direct connections between siblings, they are only connected through their parents.

As mentioned previously, some generalized association rule mining approaches use fuzzy taxonomies in order to generalize terms during the mining process, and modify support and confidence calculation based on fuzzy membership degrees. In fact, rules

referring to higher-level concepts, such as $Meat \Rightarrow Fruit$, reflect more abstract business semantics, often meaningful to decision-makers. However, specially when concerning about data semantics, the generalization strategy may lead to misunderstanding or inappropriate interpretation of the discovered knowledge. For instance, suppose we have a database that stores transactions of supermarket purchases, containing food items presented in Figure 1. Table 1 shows the contents of this database, where *Tid* is an identifier for each transaction, and *Vegetable_dishes* and *Meat* contain items bought by supermarket customers.

Table 1. Transactions of supermarket purchases.

<i>TID</i>	<i>Vegetable_dishes</i>	<i>Meat</i>
10	Apple	Chicken
20	Kaki	Turkey
30	Tomato	Chicken
40	Apple	Turkey
50	Cabbage	Sausage
60	Apple	Chicken
70	Tomato	Turkey
80	Apple	Chicken
90	Kaki	Chicken
100	Apple	Turkey

Depending on support and confidence parameters, a traditional association rule mining process can generate rules such as $Apple \Rightarrow Chicken$ and $Apple \Rightarrow Turkey$. Chen, Wei and Kerre's work proposes improvements to this process, by generalizing items according to fuzzy taxonomies that contain membership degrees. For example, they consider that one occurrence of *Chicken* or *Turkey* in a transaction also represents one occurrence of their super-item *Meat*, whereas *Sausage* represents, actually, an occurrence of value 0.6 in relation to its super-item *Meat* (see Figure 1). Hence, support and confidence calculations are modified so that generalized rules could be obtained, for instance $Apple \Rightarrow Meat$, since *Meat* is a super-item of both *Chicken* and *Turkey* items. Nevertheless, this strategy may induce to an incorrect interpretation that the *Sausage* item is also included in $Apple \Rightarrow Meat$ rule, although there is not even a transaction containing *Apple* and *Sausage* in Table 1. Therefore, although the use of fuzzy membership degree in taxonomies makes them closer to real-world domain information, it does not avoid interpretation mistakes that could be caused by generalization.

Even not comprehending fuzzy taxonomies, ExCIS [Brisson et al. 2005] adopts a strategy to avoid this problem, by generalizing an item contained in a rule only if all its siblings appear in equivalent rules. For instance, as *Chicken*, *Turkey* and *Sausage* are sub-items of *Meat* (see Figure 1), the rule $Apple \Rightarrow Meat$ would be obtained only if the mining process had generated the following rules: $Apple \Rightarrow Chicken$, $Apple \Rightarrow Turkey$ and $Apple \Rightarrow Sausage$. Although this approach prevents incorrect interpretation of rules, it may lose information about relationships among subsets of siblings, for example, that both *Chicken* and *Turkey* are related to *Apple*.

Despite of these problems, all works cited in this section have adopted the gen-

eralization strategy to filter redundant rules, regardless of the possible loss of semantics included in this process. Besides, some of these works use fuzzy logic on taxonomies in order to represent information with richer semantics. As taxonomies basically consist of *is-a* relationships, it becomes natural to incorporate fuzzy membership relations into them. On the other hand, ontologies support the representation of other semantic associations in addition to *is-a* relationships, making possible to use other fuzzy relations. Furthermore, ontologies are usually constructed by domain experts, resulting in a consensual and shared knowledge that is widely reused and accepted by many communities and applications.

Hence, in order to deal with the aforementioned generalization problem and obtain more understandable association rules, we propose an extension to the SSDM algorithm so that it can use a novel fuzzy ontology to represent semantic information on data. This ontology has fuzzy similarity relations [Zadeh 1987], making possible to express the semantic similarity between items by a similarity degree sim ($0 \leq sim \leq 1$). Such similarity relations are defined between leaf-nodes in the ontology, which represent items in the database, whereas non-leaf nodes express abstract domain concepts. Moreover, we consider that only sibling items can be semantically similar to one another, once it does not make sense to compare the semantics of non-sibling items. An example of this fuzzy ontology is presented in Figure 2, which also represents concepts and relationships of the food item domain. However, while the fuzzy taxonomy in Figure 1 contains membership degrees between concepts and its ancestors, this ontology includes similarity degrees between sibling leaf-nodes.

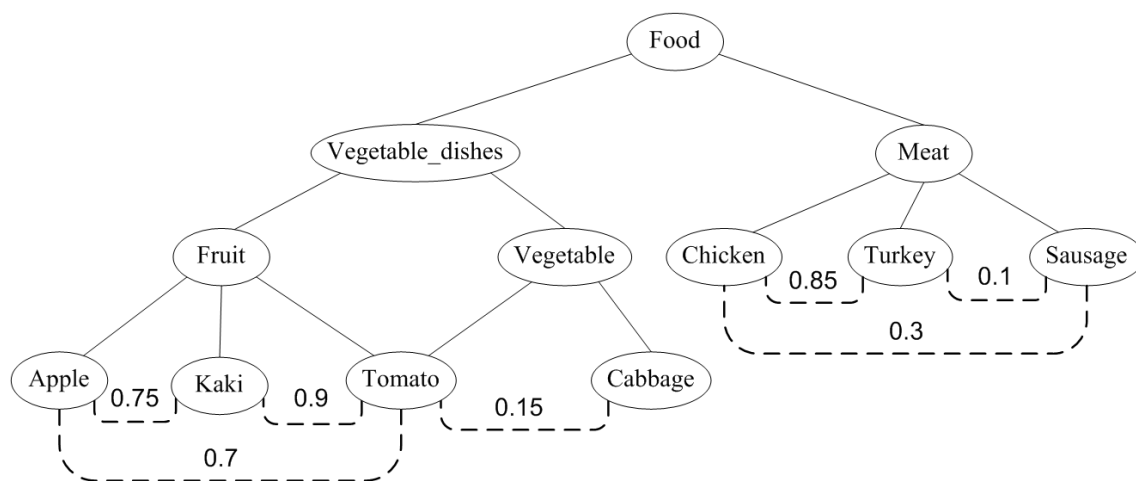


Figure 2. Fuzzy ontology of food items containing fuzzy similarity degrees.

In Figure 2, items contained in Table 1 are represented as leaf-nodes, and have semantic similarity relations (dashed lines) to their sibling items. For instance, *Apple* is similar to *Kaki* with degree 0.75 and to *Tomato* with degree 0.7. Such similarity between items is fuzzy semantic information that can be processed by the mining task. Henceforth, we show how the Extended SSDM algorithm interacts with this fuzzy ontology in order to deal with the generalization problem and obtain semantically richer association rules.

3.2. Extended SSDM

In this work, some steps from the original SSDM algorithm were modified, so that they could handle the proposed fuzzy ontology containing similarity degrees between items. Consequently, the mining process can generate more understandable and meaningful association rules, based on fuzzy background knowledge. Figure 3 shows an overview of these steps, highlighting the ones which handle the fuzzy ontology contents.

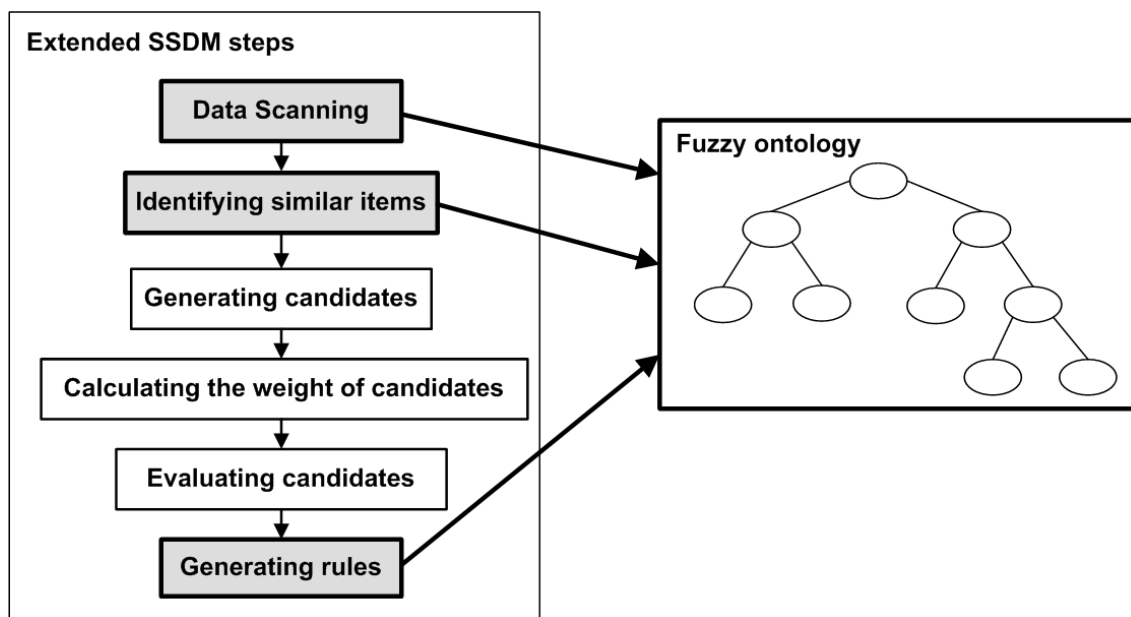


Figure 3. Extended SSDM steps.

For a better comprehension of how Extended SSDM works, we are going to use the example of supermarket purchases (Table 1) and the fuzzy ontology of food items (Figure 2). Likewise the original SSDM, the extended algorithm requires user-provided minimum support (*minsup*), minimum confidence (*minconf*) and minimum similarity degree (*minsim*) parameters. Hence, suppose that we have the following values:

- $minsup = 0.4$
- $minconf = 0.7$
- $minsim = 0.7$

Therefore, subsections 3.2.1 to 3.2.6 will describe each step, focusing on modifications made in relation to the original SSDM algorithm.

3.2.1. Data Scanning

The first step is a data scanning that identifies items in the database, generating 1-itemsets (itemsets with size one). The equivalent step in the original SSDM is responsible for classifying items according to the database column name, but now it is no longer necessary since database items have a straight correspondence to leaf-nodes of the fuzzy ontology and, consequently, similarity relations to their siblings can be easily identified. Therefore, considering the example of supermarket purchases, after data scanning we identified the following items: *Apple*, *Kaki*, *Tomato*, *Cabbage*, *Chicken*, *Turkey* and *Sausage*.

3.2.2. Identifying similar items

In the Extended SSDM, similarity degree values between sibling items are supplied by the fuzzy ontology, which specifies the semantics of the database contents. Such ontology can be created by domain experts or be reused from existing ones, since there is a crescent number of ontologies available in Semantic Web. Furthermore, pre-existent ontologies can be extended by domain specialists who can define appropriate similarity degree values, so that the mining task can handle consistent semantic information. Alternatively, it would be possible to obtain fuzzy ontologies in a semi-automatic way, as proposed by some works like [Chen et al. 2003] and [Widyantoro and Yen 2001].

Once the fuzzy ontology has been defined, this step navigates through its structure to identify semantic similarity between items. If the similarity degree between items is greater than or equal to *minsim*, a semantic similarity association is detected, meaning that items contained in this association are similar enough (and therefore interesting to the user). These pairs of items compose fuzzy associations of size 2 and are expressed by fuzzy items, where the \sim symbol is used to indicate the similarity relation between items (*item1* \sim *item2*).

After identifying fuzzy associations of size 2 (with 2 items), this step verifies the existence of similarity cycles according to the strategy proposed in [Escovar et al. 2005]. A similarity cycle is a fuzzy association of size greater than 2 that only exists if all of its items are, in pairs, sufficiently similar. As any similarity relation, a similarity cycle involves only sibling items in the fuzzy ontology. Thus, if an item belongs to more than one ancestor, it may be involved in many similarity cycles, provided that each of them contains only sibling items. For instance, *Tomato* belongs to *Fruit* and *Vegetable* (see Figure 2), and therefore *Tomato* can be involved in similarity cycles containing *Fruit* descendants and also in similarity cycles containing *Vegetable* descendants. Whereas the minimum size of a similarity cycle is 3, its maximum size is defined by the number of descendants that the ancestor of its items has. Then, the existence of similarity cycles is checked considering ancestors of identified fuzzy items, in order to obtain fuzzy associations of size k ($k \in N$, $3 \leq k \leq s$, where s is the number of descendants these ancestors have).

So, going back to the example of supermarket purchases, the fuzzy ontology of Figure 2 is analyzed and, considering *minsim* = 0.7, it is possible to identify the fuzzy items shown in Table 2. Note that one similarity cycle was identified (*Tomato* \sim *Kaki* \sim *Apple*) because all of its items are, in pairs, sufficiently similar. The notation $\text{sim}(\textit{item1}, \textit{item2})$ represents the similarity relation between *item1* and *item2*.

Table 2. Similarity relations that satisfy *minsim*.

<i>Similarity Relation</i>	<i>Fuzzy item</i>	<i>Similarity degree</i>
$\text{sim}(\textit{Tomato}, \textit{Kaki})$	<i>Tomato</i> \sim <i>Kaki</i>	0.9
$\text{sim}(\textit{Tomato}, \textit{Apple})$	<i>Tomato</i> \sim <i>Apple</i>	0.7
$\text{sim}(\textit{Kaki}, \textit{Apple})$	<i>Kaki</i> \sim <i>Apple</i>	0.75
$\text{sim}(\textit{Turkey}, \textit{Chicken})$	<i>Turkey</i> \sim <i>Chicken</i>	0.85
$\text{sim}(\textit{Tomato}, \textit{Kaki}, \textit{Apple})$	<i>Tomato</i> \sim <i>Kaki</i> \sim <i>Apple</i>	0.7

At the end of this step, we have all fuzzy associations that satisfy *minsim*, possibly including similarity cycles. Henceforth, these fuzzy items can be considered to generate rules.

3.2.3. Generating candidates

This step generates itemset candidates as well as the original SSDM does, considering items identified in the data scanning step (subsection 3.2.1) and fuzzy items obtained in the step of identifying similar items (subsection 3.2.2). After candidates were generated, they are submitted to the step of calculating correspondent weights.

3.2.4. Calculating the weight of candidates

In this step, the weight of candidates is calculated likewise it is performed in SSDM. By definition, the weight of an itemset corresponds to the number of its occurrences in the database. Thus, the database is scanned, and each of its rows is confronted with a set of itemset candidates, one after one. For each occurrence of a non-fuzzy itemset candidate in a row, its weight is incremented by 1; if the itemset candidate is fuzzy, its weight is incremented by the fuzzy weight value, given by the *Fuzzy weight* equation proposed in [Escovar et al. 2005]. Such process goes on until all rows have been scanned. After this, itemset candidates are evaluated in the next step.

3.2.5. Evaluating candidates

Since the weight of itemset candidates was calculated, we can evaluate their support in the same way SSDM does. Support corresponds to the weight divided by the number of rows (or total of transactions) in the database. If the itemset candidate is fuzzy, the fuzzy support is evaluated by simply dividing its fuzzy weight by the number of rows. Then, if the support of a candidate is greater than or equal to *minsup*, the correspondent itemset is stored in the set of frequent itemsets, which will be considered in order to generate rules. Going back to the example of supermarket purchases, the set of frequent itemsets and respective support values are presented in Table 3.

3.2.6. Generating rules

In this last step, all possibilities of antecedents and consequents are generated for each itemset belonging to frequent itemsets, likewise it is done in the original SSDM. If confidence of a rule is greater than or equal to *minconf*, then the rule is considered valid. After this, Extended SSDM checks if valid rules contain fuzzy items, in order to verify whether it is possible to generalize them. A fuzzy item can be generalized if it contains all descendants of an ancestor, according to the fuzzy ontology. For example, if we have the rule *Tomato* \sim *Kaki* \sim *Apple* \Rightarrow *Chicken*, we can generalize this fuzzy item to the ancestor *Fruit*, since all its descendants (*Tomato*, *Kaki* and *Apple*) are contained in the fuzzy item. Although we consider generalization, observe that all sub-items must be

Table 3. Frequent itemsets of the supermarket purchases database.

<i>Frequent itemset</i>	<i>Support</i>
{ <i>Chicken</i> }	0.5
{ <i>Apple</i> }	0.5
{ <i>Turkey</i> }	0.4
{ <i>Tomato</i> ~ <i>Apple</i> }	0.595
{ <i>Kaki</i> ~ <i>Apple</i> }	0.6125
{ <i>Turkey</i> ~ <i>Chicken</i> }	0.8325
{ <i>Tomato</i> ~ <i>Kaki</i> ~ <i>Apple</i> }	0.765
{ <i>Turkey</i> ~ <i>Chicken</i> , <i>Apple</i> }	0.4625
{ <i>Turkey</i> ~ <i>Chicken</i> , <i>Tomato</i> ~ <i>Apple</i> }	0.5503
{ <i>Turkey</i> ~ <i>Chicken</i> , <i>Kaki</i> ~ <i>Apple</i> }	0.5665
{ <i>Tomato</i> ~ <i>Kaki</i> ~ <i>Apple</i> , <i>Chicken</i> }	0.425
{ <i>Turkey</i> ~ <i>Chicken</i> , <i>Tomato</i> ~ <i>Kaki</i> ~ <i>Apple</i> }	0.7076

included in the rule. That is why the fuzzy item *Turkey* ~ *Chicken* can not be generalized to *Meat*, as it does not contain all *Meat* descendants. Therefore, Extended SSDM expresses semantic similarity between items in generated rules, and performs generalization only when it is appropriate, avoiding incorrect knowledge interpretation. Moreover, in case generalization is not performed, Extended SSDM does not lose information about relationships among subsets of siblings, because it analyzes semantic similarity relations between them.

As soon as this verification is concluded, all valid rules are exhibited to the user. Antecedents and consequents of a rule may contain fuzzy items, whose values of support and confidence reflect the influence of the similarity degree between items. Besides showing antecedent, consequent, support and confidence of each rule, Extended SSDM exhibits all ancestors of items contained in rules, making them more understandable to users. Consequently, it is possible to analyze the context in which rules are included and give a better support to knowledge discovery. Considering the example of supermarket purchases, Figure 4 illustrates how Extended SSDM shows generated rules.

As previously mentioned in subsection 3.1, existent approaches would have inadequately generalized the rule $Apple \Rightarrow Turkey \sim Chicken$ to $Apple \Rightarrow Meat$ or else would have not considered the semantic relation between *Turkey* and *Chicken* in the generated rules. Instead, Extended SSDM can consider similarity between sibling items (*Turkey* ~ *Chicken*) and also can obtain correct generalizations like the rule $Chicken \Rightarrow Fruit$, since all *Fruit* descendants are part of the similarity cycle (*Tomato* ~ *Kaki* ~ *Apple*) that originated this rule.

4. Experiments

We performed some experiments with Extended SSDM, considering real data from the Brazilian Demographic Census 2000, provided by IBGE¹ (*Brazilian Institute of Geography and Statistics*). Two datasets were analyzed: *IBGE1*, containing information about

¹<http://www.sidra.ibge.gov.br/cd/>

```

Food > Vegetable_dishes > Vegetable > Tomato
Food > Vegetable_dishes > Fruit > Tomato
Food > Vegetable_dishes > Fruit > Apple
Food > Vegetable_dishes > Fruit
Food > Meat > Turkey
Food > Meat > Chicken

Rules generated (minsup=0.4, minconf=0.7, minsim=0.7)

Tomato~Apple->Turkey~Chicken sup=0.55037504 conf=0.925

Kaki~Apple->Turkey~Chicken sup=0.56656253 conf=0.925

Apple->Turkey~Chicken sup=0.4625 conf=0.925

Fruit->Turkey~Chicken sup=0.70762503 conf=0.9250001

Turkey~Chicken->Fruit sup=0.70762503 conf=0.8500001

Chicken->Fruit sup=0.425 conf=0.85

```

Figure 4. Rules generated by Extended SSDM.

Years of study, Sex and Race or ethnicity; and IBGE2, which relates Race or ethnicity and Conjugal state.

First of all, it was necessary to obtain a fuzzy ontology that provides semantic similarity relations between items. Thus, after having analyzed the data and the domain of demographic characteristics, we created the fuzzy ontology presented in Figure 5, which contains fuzzy similarity degrees between some items belonging to *Race_or_ethnicity* and *Conjugal_state* concepts. Such ontology was modeled with *Protégé-OWL* [Knublauch et al. 2004], a tool for editing ontologies in OWL (*Web Ontology Language*). Moreover, we used the *Jena Framework* [Carroll et al. 2004] to support navigation through ontology concepts and relationships, making Extended SSDM able to obtain similar items and correspondent similarity degrees.

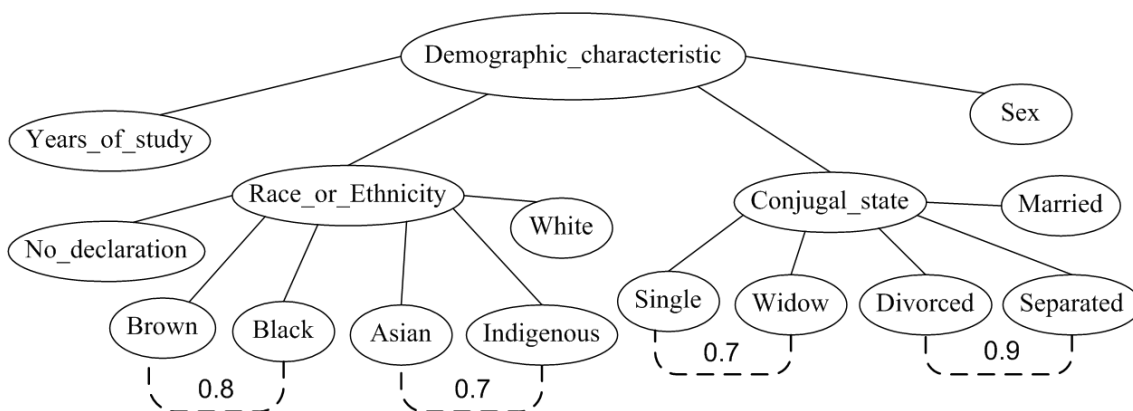


Figure 5. Fuzzy ontology of demographic characteristics.

Once we have the fuzzy ontology of demographic characteristics (Figure 5), *IBGE1* and *IBGE2* datasets can be mined considering semantic similarity between items.

We start testing *IBGE1*, which contains 9,997 transactions, with parameter values $minsup = 0.19$, $minconf = 0.5$ and $minsim = 0.8$. In this first test, semantic similarity between items belonging to *Race_or_ethnicity* is reflected in generated rules shown in Figure 6.

```

Demographic_characteristic > Years_of_study > Four_to_ten_years
Demographic_characteristic > Race_or_ethnicity > White
Demographic_characteristic > Race_or_ethnicity > Brown
Demographic_characteristic > Race_or_ethnicity > Black
Demographic_characteristic > Sex > Male
Demographic_characteristic > Sex > Female

Rules generated (minsup=0.19 minconf=0.5 minsim=0.8)

Brown~Black->Male sup=0.19697909 conf=0.5025264

Four_to_ten_years->Female sup=0.24707413 conf=0.5059402

Female->White sup=0.28978693 conf=0.56350905

White->Female sup=0.28978693 conf=0.52778286

Male->White sup=0.2592778 conf=0.53377265

Four_to_ten_years->White sup=0.2706812 conf=0.554281

```

Figure 6. Extended SSDM results for IBGE1 dataset.

Our second test involved *IBGE2*, which contains 10,001 transactions. This time, both *Race_or_ethnicity* and *Conjugal_state* have semantically similar sub-items, which are considered in the mining task. Assuming parameter values $minsup = 0.19$, $minconf = 0.5$ and $minsim = 0.7$, Extended SSDM obtain the rules presented in Figure 7.

```

Demographic_characteristic > Race_or_ethnicity > White
Demographic_characteristic > Race_or_ethnicity > Brown
Demographic_characteristic > Race_or_ethnicity > Black
Demographic_characteristic > Conjugal_state > Widow
Demographic_characteristic > Conjugal_state > Single
Demographic_characteristic > Conjugal_state > Married

Rules generated (minsup=0.19 minconf=0.5 minsim=0.7)

Married->White sup=0.22467753 conf=0.60680526

Brown->Single sup=0.23327667 conf=0.618177

Brown~Black->Single sup=0.24603538 conf=0.6174345

Brown->Widow~Single sup=0.21009898 conf=0.55675673

Brown~Black->Widow~Single sup=0.2223633 conf=0.5580285

```

Figure 7. Extended SSDM results for IBGE2 dataset.

Observe that some generated rules contain fuzzy items, for instance the rules $Brown \sim Black \Rightarrow Male$ (Figure 6) and $Brown \Rightarrow Widow \sim Single$ (Figure 7). Such rules contain interesting information, and were only discovered because the mining

process considered the semantic similarity between items, according to the strategy proposed in the original SSDM. For example, *Brown* and *Black* have only appeared among rules because they were considered similar to each other. Likewise, *Widow* would not have been contained in rules unless it were considered similar to *Single*.

Differently from aforementioned approaches, *Race_or_ethnicity* descendants, for example, are not generalized because such generalization could lead to an incorrect interpretation that *Asian*, *Indigenous* and *No_declaration* items are also included in rules. Moreover, rules obtained by Extended SSDM reflect semantic similarity between subsets of sibling items even when generalization is not performed. This is the case of similar relations $Brown \sim Black$ and $Widow \sim Single$, which would not have appeared in rules if they were respectively generalized to *Race_or_ethnicity* and *Conjugal_state*. Therefore, it is possible to reveal more relevant information thanks to the proposed strategy.

5. Conclusions

In this work, we extended the SSDM algorithm in order to use ontologies as background knowledge to represent semantics over the mined data. Therefore, Extended SSDM can reuse consensual and shared knowledge, easing the process of acquiring semantic information. As SSDM considers uncertainty on data, fuzzy logic concepts are incorporated into ontologies, resulting in fuzzy ontologies that can capture richer semantics than crisp ontologies. However, differently from existent approaches that traditionally incorporate membership degrees to *is-a* relationships, we proposed a fuzzy ontology that include similarity degree values between concepts. In addition, such fuzzy ontologies can be further used in Semantic Web mining.

Consequently, Extended SSDM can process the proposed fuzzy ontology in order to generate semantically richer rules, meanwhile avoiding incorrect interpretation of improperly generalized rules. This is because a more suitable generalization is performed, considering that all concept descendants must be included in rules. Moreover, in case generalization is not performed, Extended SSDM does not lose information about relationships among subsets of siblings, because it analyzes semantic similarity relations between them. Another important contribution is that Extended SSDM exhibits all ancestors of items contained in rules, besides showing antecedent, consequent, support and confidence of each rule. Hence, it is possible to analyze the context in which rules are included and give a better support to knowledge discovery.

6. Acknowledgments

We would like to thank Capes/Brazil for supporting this work.

References

- Agrawal, R. and Srikant, R. (1994). Fast Algorithms for Mining Association Rules in Large Databases. In *20th International Conference on Very Large Data Bases*, pages 487–499, Santiago de Chile, Chile.
- Brisson, L., Collard, M., and Pasquier, N. (2005). Improving Knowledge Discovery Process Using Ontologies. In *International ICDM Workshop on Mining Complex Data*, Houston, Texas, USA.

- Carroll, J. J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A., and Wilkinson, K. (2004). Jena: implementing the semantic web recommendations. In *International World Wide Web Conference*, pages 74–83, New York, NY, USA. ACM Press.
- Chen, G. and Wei, Q. (2002). Fuzzy association rules and the extended mining algorithms. *Information Sciences - Informatics and Computer Science*, 147(1-4):201–228.
- Chen, G., Wei, Q., and Kerre, E. E. (2000). Fuzzy Data Mining: Discovery of Fuzzy Generalized Association Rules. In Bordogna, G. and Pasi, G., editors, *Recent Issues on Fuzzy Databases*, pages 45–66. Physica-Verlag.
- Chen, X., Zhou, X., Scherl, R. B., and Geller, J. (2003). Using an Interest Ontology for Improved Support in Rule Mining. In *5th International Conference on Data Warehousing and Knowledge Discovery (DaWaK)*, pages 320–329, Prague, Czech Republic.
- Escovar, E. L. G., Biajiz, M., and Vieira, M. T. P. (2005). SSDM: A Semantically Similar Data Mining Algorithm. In *XX Simpósio Brasileiro de Banco de Dados (SBBD)*, pages 265–279, Uberlândia, MG, Brasil.
- Hong, T.-P., Lin, K.-Y., and Wang, S.-L. (2003). Fuzzy data mining for interesting generalized association rules. *Fuzzy Sets Systems*, 138(2):255–269.
- Hou, X., Gu, J., Shen, X., and Yan, W. (2005). Application of Data Mining in Fault Diagnosis Based on Ontology. In *Third International Conference on Information Technology and Applications (ICITA'05)*, pages 260–263, Sydney, Australia.
- Knublauch, H., Ferguson, R. W., Noy, N. F., and Musen, M. A. (2004). The Protégé OWL Plugin: An Open Development Environment for Semantic Web Applications. In *Third International Semantic Web Conference (ISWC)*, pages 229 – 243, Hiroshima, Japan.
- Parry, D. (2004). Fuzzification of a standard ontology to encourage reuse. In *IEEE International Conference on Information Reuse and Integration (IRI)*, pages 582 – 587, Las Vegas, NV, USA.
- Pohle, C. (2003). Integrating and Updating Domain Knowledge with Data Mining. In *VLDB PhD Workshop*, Berlin, Germany.
- Quan, T. T., Hui, S. C., and Cao, T. H. (2004). FOGA: A Fuzzy Ontology Generation Framework for Scholarly Semantic Web. In *ECML/PKDD Workshop on Knowledge Discovery and Ontologies*, pages 37–48, Pisa, Italy.
- Shekar, B. and Natarajan, R. (2004). A Framework for Evaluating Knowledge-Based Interestingness of Association Rules. *Fuzzy Optimization and Decision Making*, 3(2):157–185.
- Srikant, R. and Agrawal, R. (1995). Mining Generalized Association Rules. In *21th International Conference on Very Large Data Bases*, pages 407–419, Zurich, Switzerland.
- Wallace, M. and Avrithis, Y. (2004). Fuzzy relational knowledge representation and context in the service of semantic information retrieval. In *IEEE International Conference on Fuzzy Systems*, volume 3, pages 1397– 1402, Budapest, Hungary.
- Widyantoro, D. and Yen, J. (2001). A fuzzy ontology-based abstract search engine and its user studies. In *10th IEEE International Conference on Fuzzy Systems*, pages 1291–1294, Melbourne, Vic., Australia.

Zadeh, L. A. (1987). Similarity Relations and Fuzzy Orderings. In *Fuzzy Sets and Applications: Select Papers by L. A. Zadeh*, pages 81–104. Wiley-Interscience.