Appl Intell (2012) 36:857–869
DOI 10.1007/s10489-011-0301-4

ORIGINAL PAPER

Ontology-based user profile learning

Victoria Eyharabide · Analía Amandi

Published online: 3 June 2011

Abstract Personal agents gather information about users in a user profile. In this work, we propose a novel ontology-based user profile learning. Particularly, we aim to learn context-enriched user profiles using data mining techniques and ontologies. We are interested in knowing to what extent data mining techniques can be used for user profile generation, and how to utilize ontologies for user profile improvement. The objective is to semantically enrich a user profile with contextual information by using association rules, Bayesian networks and ontologies in order to improve agent performance. At runtime, we learn which the relevant contexts to the user are based on the user's behavior observation. Then, we represent the relevant contexts learnt as ontology segments. The encouraging experimental results show the usefulness of including semantics into a user profile as well as the advantages of integrating agents and data mining using ontologies.

Keywords Ontology segmentation · User profile · Association rules · Bayesian network · Spreading activation

V. Eyharabide (✉) · A. Amandi
ISISTAN Research Institute, Computer Science Department, Unicen University, Tandil, Argentina
e-mail: veyharab@exa.unicen.edu.ar

A. Amandi
e-mail: amandi@exa.unicen.edu.ar

V. Eyharabide · A. Amandi
CONICET, Consejo Nacional de Investigaciones Científicas y Técnicas, Buenos Aires, Argentina

1 Introduction

Personal agents gather information about users in a user profile. Once the user profile is defined, the agent can use, learn, and infer new knowledge based on this profile to personalize the assistance offered to the user. By improving the content of the user profile, the agent improves its performance. In this work, we use data mining techniques enriched with ontological knowledge to improve the semantics of the user profile.

There are several representations for user profiles [1, 2]. However, in the personal agents found in the literature, the *type* of information stored in the user profile is static, it is determined during design time and it cannot be updated or modified at execution time. For example, if the agent developer decides on a user profile containing web page URLs visited by the user or keywords extracted from email bodies, it is not possible to store any other type of information afterwards.

To understand why a user profile should have dynamic types of information, consider a personal agent that selects digital news articles according to the user's preferences. Imagine three different users who read news articles about football. While the first user reads news articles about football in general (his/her interest is the complete football section), the second user is a fan of a particular football player (his/her interest is the football player whose name appears in a news article body regardless of the news article section), and the third user reads news articles written by certain football columnist (his/her interest is the journalist who writes the news article). Although the three users are interested in football, they have completely different interests: a newspaper section, a word in the news article body and a news article author respectively. Consequently, when storing user's preferences in a user profile, it is not enough to gather some keywords extracted from the news article body. On the contrary, first, it is necessary to learn, among the context attributes of a news article, which one determines the user's decision of reading it (whether it is the section, the content or the author) and then, gather the context attribute values in the user profile ("football section", "Maradona" or the journalist "John Smith"). User's preferences vary not only from user to user but also for the same user depending on the context. The previous examples are reading preferences; however, those users could have different preferences to comment on or forward news articles. Consequently, the agent should learn (i) the user's preferences (as regards reading, commenting or forwarding) and (ii) the context in which he/she prefers them. Therefore, it is vital to include contextual information in the user profile.

Generally, users consider only a small part of the context as relevant [3]. As shown in the examples, not all the contextual attributes of a news article influence the user's reading preferences. However, most proposals analyzed consider all the contextual attributes as relevant to all the user's preferences. Conversely, in this approach we learn which the relevant context is for each user's preference. In order to determine relevant context, we consider it necessary to learn: (i) the *type* of contextual variables that determine the user's preference (in the example: section, content, author) and (ii) the *values* of the contextual variables (Football, Maradona, John Smith). To use only a reduced amount of context not only simplifies the subsequent reasoning mechanisms but also contributes to the acquisition of better results. Next-generation user profiles should be not only a passive repository of user data, but also an active component able to learn and update both the information and the type of information stored.

Nowadays, what is relevant context in an application is determined (i) initially by the agent developer at design time; or (ii) later by the user (generally through explicit feedback). We are convinced that which contextual information is relevant or not cannot be determined by the agent developer beforehand, because it is impossible to anticipate the preferences of all possible users due to the great variety of interests and the heterogeneous application domains in which the agent can assist a user. Therefore, there are two possible ways to detect relevant context attributes among all the available context attributes: to infer them through the user-application interaction or to explicitly inquire the user. On the one hand, asking the user is the simplest way to obtain that information. However, it is tedious for users to give explicit feedback because it is time consuming and it involves quite an effort. Even more, the user may not want to give that information, not even know what the context is, or not know how to express it accurately [1]. On the other hand, it is complex to infer relevant context through the user's behavior observation, since high-level context attributes should be found by starting from the low-level context attributes obtained from the user-application interaction. Despite that complexity and in order to avoid inquiring the user, our goal is to learn which context is relevant to the user during execution time on the basis of the user's behavior observation.

In this work, we take advantage of data mining techniques enriched with the semantics gathered within an ontology to produce highly relevant user profiles. Our objective is to personalize the user profiles with contextual information in order to improve the agent's performance by combining data mining and ontologies. We enrich the user profile semantically with an ontological representation of context to dynamically adapt the attribute selection according to the user's preferences. In particular, we learn which concepts and relations of a context ontology are relevant to the user for a certain preference. Initially, by using data mining techniques we identify patterns in the user's transaction data describing which context attributes frequently occur together. Then, we look for the common semantics to those context attributes using an ontology.

The paper is organized as follows. We first begin by studying different user profile representation approaches and presenting an example to clarify our proposal jointly with some definitions our approach is based on. We then explain our ontology-based user profile approach by explaining each of its building stages. Later, we depict a case study in which our user profiling technique is used to personalize the recommendation of digital news articles. Afterwards, we show the results obtained from the experiments we carried out in order to validate our approach. Finally, we conclude by analyzing the importance of using semantic-enriched data mining techniques for effective and efficient deployment of personal agents.

2 Related work in user profile representations

Personal agents do not usually include context information in user profiles [4]. Only some agents [1, 2] which consider a limited version of context (such as time and the user's physical location) model the user profile more dynamically. In the analyzed proposals, context is (i) defined informally, (ii) generally known in advance and (iii) determined in a fixed way. Usually, they apply machine learning techniques to isolated attributes and they do not analyze the semantic relations among attributes.

A way to introduce semantics in a user profile is by using ontologies. An ontology [5] is a formal representation of knowledge as a set of concepts within a domain, and the relationships between those concepts. Nowadays, given their powerful knowledge representation formalism and associated inference mechanisms, ontologies are emerging as a natural choice for the next-generation user profiles [6]. Ontology-based user profile representations [7–9] vary in ontology type, purpose, domain and way the ontologies are used. The main advantages of ontology-based user profiles are: (a) they favor the interpretation of the information gathered in the user profile, (b) they facilitate the exchange and reutilization of user profiles among different systems, (c) they overcome the current syntactic and structural differences among different proposals of user modeling and (d) they structure the domain knowledge of the application in which the agent assists the user.

There are several related works that automatically create user profiles based on ontologies. In particular, some of them structure the user profile as a weighted concept hierarchy. Mylonas et al. [10] combine contextualization and personalization methods to improve the performance of personalized information retrieval. A similar aspect with our approach is the use of semantic concepts for the representation of contextual meanings; whereas a key differentiating point is the techniques used to tackle the uncertainty involved in context learning: they use fuzzy set while we use spreading activation. Gauch et al. [11] create ontology-based user profiles with the aim of using them for personalization purposes. In this approach, the authors classify web pages into the appropriate concept(s) in a reference ontology using a vector-space classifier. However, their work specifies only "super-class" and "sub-class" relations. Tao et al. [6, 12] describe a knowledge-based model for ontology mining. This work analyzes the relations among the subjects in the ontology for user profile acquisition and the capture of user information needs. Sieg et al. [13] present a personalized search that involves building ontology-based user profiles of context by assigning interest scores to existing concepts in a domain ontology. As in our proposal, this approach uses a spreading activation algorithm to maintain the interest scores based on the user's behavior; however, its "ontology" is only a simple hierarchy of topics. In contrast, our approach integrates data mining techniques based on association rules mining, Bayesian nets, spreading activation and ontologies into the learning process of user profiles.

Next-generation personal agents cannot define context statically. Users execute different tasks at the same time, they move from one place to another and they constantly modify their preferences. In these fluctuating scenarios, the user's context is extremely dynamic [14]. Therefore, it is practically impossible to define context in a fixed way in such changing environments. To overcome these drawbacks, we propose the use of data mining techniques enriched with ontological knowledge to define context explicitly in order to include it in a user profile. Our proposal contrasts with traditional user profiles approaches, some of which are illegible and merely passive data repositories. By profiting from the human-readable characteristic of ontologies, we enable the user not only to visualize his/her user profile, but also to understand, modify, and update it afterwards.

3 Including context in the user profile

To clarify our proposal, we will continue with the example of a personal agent that selects digital news articles according to the user's preferences. Suppose the user has read a news article. Why has he/she read it? Has he/she read it because it was written by a certain journalist? Is the topic of the news article of his/her interest? Does the news article mention a partner company the user works in? When a user executes a task, we should find, among the contextual variables associated to that task, which ones influence the user's execution. We represent the context variables as concepts of an ontology and the relations among context variables as relations among ontological concepts. Consequently, we should find which concepts and relations of the ontology are relevant to the user when he/she executes a task.

Consider the context variables of the ontology in Fig. 1 and two users: Mary and John. Suppose Mary avoids reading news articles with figures of speech and prefers precise descriptions to colloquial terms. Therefore, she is interested in news articles written by journalists who have a neutral writing style. In this case, the author's writing style determines whether Mary will read a news article or not. Thus, in Mary's user profile we only relate *News* with *Author* and *Writing Style*, discarding the remaining context variables of *News*. In Fig. 1(a) Mary's user profile shows that she reads news articles written by three journalists which have a neutral writing style.

In contrast, imagine that John reads the most read news articles in the sports section (usually digital newspapers indicate which of the news articles are read the most). He decides to read sports news articles ranking high in daily publications reading regardless of the author. John's user profile is depicted in Fig. 1(b).

At this point, it is important to address the issue of a possible invasion of privacy by this intelligent profile creation. This technique has the potential of creating a user profile far more extensive than a user might desire. Therefore, we always ask for the user's permission before building his/her profile. Even more, as the resultant user profiles will be used to give personalized assistance to the user, he/she will be the only one with access to her/his profile's information.

In this work, we extract relevant segments out of a large ontology with the purpose of building semantically enriched user profiles. We assume that the application domain in which we learn the user profiles is modeled in one or several ontologies. In [15], we presented a multi-dimensional user profile model based on ontologies. Herein, we explain

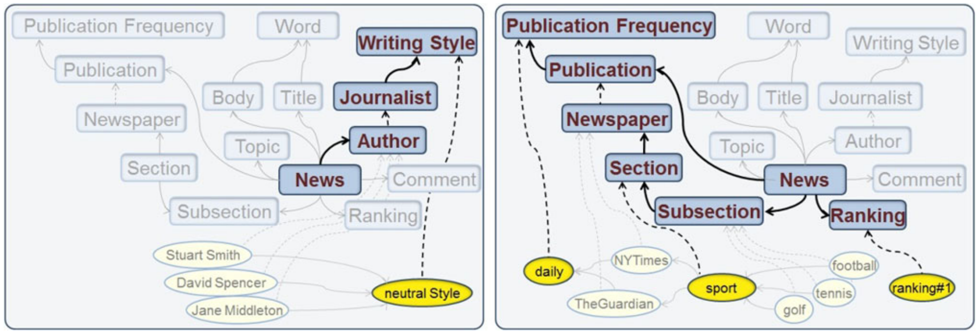


Fig. 1 User profile of (a) Mary and (b) John respectively

the user profile learning process considering only one ontology. However, the proposed approach can be easily extended to several ontologies by applying our technique in each of them. In particular, we select the concepts and relations that are relevant to the user from a domain ontology by using data mining and ontology segmentation techniques [16]. An ontology segmentation technique takes advantage of the semantic relations between ontological concepts in order to create custom ontologies. A segment is more than a simple subset of concepts and relations; it is an ontology in itself but smaller than the original one. That is to say, all the components of a segment (concepts, relations, axioms and instances) are part of the ontology from which that segment was extracted. We represent a user profile as an ontological segment personalized according to the user's preferences. Particularly, we learn which *context attributes* are relevant to the user for each possible user's *action*. In the example, the actions are those that the user can execute on a news article (such as *read*, *save*, *print*, *e-mail* or *comment on*) and the context attributes are the concepts in the domain ontology (such as section, author, or title). The news article attributes correspond to ontology concepts and the news article attribute values correspond to instances of those concepts. In an ontology, a concept is usually related to one or more concepts. Therefore, by navigating through the ontological relations, we search for concepts that are shared among several news article attributes. This procedure is shown in Fig. 1. Initially, by using data mining techniques, we identify patterns in the user's transaction data describing which news article attributes frequently occur together. Then, as news article attributes are represented by instances in the ontology, we find out if those news article attributes are related to some shared instance (ovals in Fig. 1). Finally, we extract relevant ontological segments from the shared instances discovered (squares in Fig. 1).

4 Learning the relevant contexts for the user

To find relevant contexts, we focus on the user's preferences as one of the several context information examples that can be gathered in the user profile. As we mentioned previously, the user's preferences change according to context. Therefore, we define a *Preference* as an *Action* that the user prefers in a given *Context*, a *Relevance* that indicates how much the user prefers to perform the action, a *Certainty* representing how certain the agent is about this user's preference and a *Date* denoting when that preference was obtained:

$$\text{Preference} = \{\text{Action, Context, Relevance, Certainty, Date}\}$$

A user can prefer to perform the same action in different *contexts*. For example, Thomas can *comment* on news articles that are *in the technology section whose title contains the word Google* or that are *about global warming*. Therefore, we can have several *preferences* with the same *action* but in different *contexts*. In this work, we consider context as the ontology segment that is relevant to action a. As we mentioned above, an ontology segment is composed of the concepts and relations of the ontology that the user considers relevant. Consequently, we define *Context(a)* as the set union of the sets *RelevantConcepts(a)* and *RelevantRelations(a)*:

$$\text{Context}(a) = \text{RelevantConcepts}(a) \cup \text{RelevantRelations}(a)$$

The predicate *isRelevantConcept(c, a)* states that concept c is relevant to action a. If *OntoConcepts* is the set of all the concepts of an ontology, we call *RelevantConcepts(a)* the subset of *OntoConcepts* which defines the concepts that the user considers relevant for the action a (we describe algorithms for these functions later in the paper):

$$\text{RelevantConcepts}(a) = \{c \in \text{OntoConcepts} \mid \text{isRelevantConcept}(c, a) = \text{TRUE}\}$$

Similarly, we define *isRelevantRelation(r, a)*, a predicate stating that relation r is relevant to action a. Being *OntoRelations* the set of all ontology relations, we call *RelevantRelations*(a) the subset of *OntoRelations* that defines the set of relevant relations to action a. A relation is relevant to action a if the predicate *isRelevantRelation*(r, a) is true and if the relation connects two concepts belonging to the set *RelevantConcepts*(a):

$$\begin{aligned}&\text{RelevantRelations}(a)\\&\quad=\{r\in\text{OntoRelations}\mid\text{isRelevantRelation}(r,a)\\&\qquad=\text{TRUE}\wedge\text{relates}(r,c_1,c_2)\\&\qquad=\text{TRUE}\wedge c_1,c_2\in\text{RelevantConcepts}(a)\}\end{aligned}$$

To clarify these ideas, consider Thomas' example. As we mentioned above, Thomas prefers to comment on news articles in two different cases. Therefore, there are two *preferences* for action $a =$ "comment": one preference for $Context_1$ and another preference for $Context_2$. While the first context $Context_1$ describes news articles *in the technology section whose title contains the word Google*; the second context $Context_2$ describes news articles *about global warming*. In $Context_1$, relevant concepts are "News", "Section", "Title" and "Word"; and the relevant relations are "hasSection", "hasTitle" and "contains". In $Context_2$, relevant concepts are "News" and "Topic"; and "isAbout" is the relevant relation. In turn, each user's preference has different levels of relevance and certainty values, as well as a different date.

$$\begin{aligned}\text{Context}_1(\text{comment})&=\mathit{RelevantConcepts}_1(\text{comment})\\&\quad\cup\mathit{RelevantRelations}_1(\text{comment})\end{aligned}$$

$$\begin{aligned}&\text{RelevantConcepts}_1(\text{comment})\\&\quad=\{\text{News, Section, Title, Word}\}\end{aligned}$$

$$\begin{aligned}&\text{RelevantRelations}_1(\text{comment})\\&\quad=\{\text{hasSection, hasTitle, contains}\}\end{aligned}$$

$$\begin{aligned}\text{Context}_2(\text{comment})&=\mathit{RelevantConcepts}_2(\text{comment})\\&\quad\cup\mathit{RelevantRelations}_2(\text{comment})\end{aligned}$$

$$\text{RelevantConcepts}_2(\text{comment})=\{\text{News, Topic}\}$$

$$\text{RelevantRelations}_2(\text{comment})=\{\text{isAbout}\}$$

In turn, a context can be instantiated. Therefore, we define *InstContext(a)* as the union of the sets *InstRelevantConcepts(a)* and *RelevantRelations(a)*:

$$\begin{aligned}\text{InstContext}(a)&=\mathit{InstRelevantConcepts}(a)\\&\quad\cup\mathit{RelevantRelations}(a)\end{aligned}$$

We define the instantiation of concept c as the tuple (c, i), where i is an instantiation of c in a given moment. For instance, the tuple (Person, *Mary*) indicates that *Mary* is an instantiation of the concept *Person.* We call *OntoInstantiations*(c) the set of all the instances of concept c in the ontology. Finally, we define *InstRelevantConcepts*(a) as the set of instantiated context concepts relevant for action a:

$$\begin{aligned}&\text{InstRelevantConcepts}(a)\\&\quad=\{(c,i)\mid c\in\mathit{RelevantConcepts}(a)\\&\qquad\wedge i\in\mathit{OntoInstantiations}(c)\}.\end{aligned}$$

For the same action, different users could not only consider different contexts but also instantiate them in different ways. For example, for Thomas we instantiate $RelevantConcepts_1$ as $InstRelevantConcepts_{11}$ and $RelevantConcepts_2$ as $InstRelevantConcepts_{21}$ respectively:

$$\begin{aligned}&\text{InstRelevantConcepts}_{11}(\text{comment})\\&\quad=\{(\text{Section},\mathit{technology}),(\text{Word},\mathit{Google})\}\end{aligned}$$

$$\begin{aligned}&\text{InstRelevantConcepts}_{21}(\text{comment})\\&\quad=\{(\text{Topic},\mathit{global\_warming})\}\end{aligned}$$

However, for user Williams we can instantiate $RelevantConcepts_1$ as $InstantiatedRelevantConcepts_{12}$ indicating that he comments on *sport news articles that contain the word "Tiger Woods" in the title*:

$$\begin{aligned}&\text{InstRelevantConcepts}_{12}(\text{comment})\\&\quad=\{(\text{Section},\mathit{sport});(\text{Word},\mathit{Tiger\_Woods})\}\end{aligned}$$

In contrast, a third user, Anne, may not care about the section, the title or the topic when commenting on news articles. Imagine Anne only comments on news articles written by the journalist "John Smith". In this case, her context is $Context_3$(comment) and we instantiate the relevant concepts $RelevantConcepts_3$(comment) as $InstRelevantConcepts_{31}$(comment):

$$\begin{aligned}\mathit{Context}_3(\text{comment})&=\mathit{RelevantConcepts}_3(\text{comment})\\&\quad\cup\mathit{RelevantRelations}_3(\text{comment})\end{aligned}$$

$$\text{RelevantConcepts}_3(\text{comment})=\{\text{News, Author}\}$$

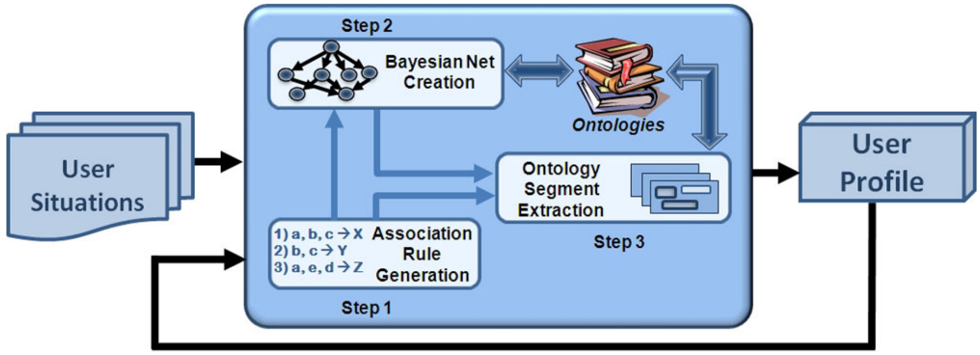
$$\text{RelevantRelations}_3(\text{comment})=\{\text{hasAuthor}\}$$

$$\begin{aligned}&\text{InstRelevantConcepts}_{31}(\text{comment})\\&\quad=\{(\text{Author},\mathit{John\_Smith})\}\end{aligned}$$

5 User profiling technique

By using data mining techniques enriched with ontological knowledge, we learn the relevant context and its instantiation for each user's preference. Our user profiling technique has two main phases. In the first phase, we generate association rules from the user logs to find the most frequent attributes in data. The most frequent attributes are supposed to

Fig. 2 Proposed user profiling technique



be the most relevant to the user. However, simple attributes are usually too specific to detect high-level context. Consequently, in the second phase, we look for what those attributes have in common using the semantics of domain ontologies. Once we have discovered the shared concept of the attributes, we extract the ontology segments that represent the relevant context to the user. Figure 2 depicts the main components of our user profiling technique.

We build the user profile using three iterative steps: (1) Association rule generation; (2) Bayesian Net Creation; and (3) Ontology Segment Extraction. These steps are explained in detail in the following sections.

Step 1: Association rule generation

Initially, we generate association rules to identify relations among, in our example, news and actions that the user has executed on them (such as *read carefully*, *read superficially*, *comment on*, *save*, *print*, *email*, *vote* or *blog*). Therefore, we obtain the most relevant news article attributes in training data for each user action. For example, given a certain attribute (such as the author) an association rule indicates if the user reads that news article.

We extract association rules from a collection of *user situations* using the Apriori algorithm [17]. A user situation is a news article and the user's action on it. To apply association rule mining, each user situation is considered as a transaction in which each news article attribute is seen as an item. Then, we group those rules that have the same user action in its consequent. As a result, each association rule group describes the news article on which the user has performed the same action. Particularly, in a group the antecedent of each rule is a set of attributes describing a certain news article, and the consequent of the rule is the action that the user has executed on it.

A large number of association rules is obtained when applying data mining techniques to a training set. Usually, we require some pruning and summarizing methods in order to extract meaningful patterns. Consequently, we filter out redundant and uninteresting rules in each group using (a) some well-known pruning heuristics [18] and (b) some new ontology-based pruning heuristics [19].

Step 2: Bayesian Net Creation

Up to this point, we have only considered news article attributes. Although these attributes have important meanings, they are too specific to represent high-level relevant context to the user. We can note this when looking that some of the obtained rules are very similar (rules having the same antecedent or consequent or only differing in an attribute or specific value). This leads to the assumption that those rules might have "something" in common. For example, although two *news articles* can be different, they can be written by the same *journalist*. In turn, different *journalists* can work in different *journals* which belong to the same *holding company*. Consequently, by using domain ontologies we try to find the common semantics of similar rules. Particularly, we search into the ontology for what the different attributes have in common. Herein, contextual attributes are represented as ontological concepts. Therefore, we should find the common semantics among those concepts. However, the idea is not only to find out if attributes detected by the rules are related to common instances in the ontology, but also if they are related by the use of the same types of ontological relations. Thus, in this step we look for the common relations among ontological concepts. To achieve this, we create Bayesian Nets in which the evidence is added to individual nodes and some nodes become more probable than others [20].

We use the K2 algorithm [21] to learn the Bayesian Nets structure and the Bucket Elimination algorithm [22] to make inferences over those nets. To create a Bayesian Network, we follow the ontological relations in order to extract all the ontological paths that begin in a contextual attribute detected by the association rules (as mentioned above, contextual attributes are represented by ontological concepts).

To clarify these ideas, consider the example presented in Fig. 3. Suppose that "*News1*" and "*News2*" are two contextual attributes detected by the rules. There are two possible ontological paths starting from "*News1*": path1 and path2; whereas "*News2*" gives path3 and path4 (Fig. 3(a)). All the ontological paths obtained are organized in a file which is the input for the K2 algorithm [21]. As seen in Fig. 3(b), in

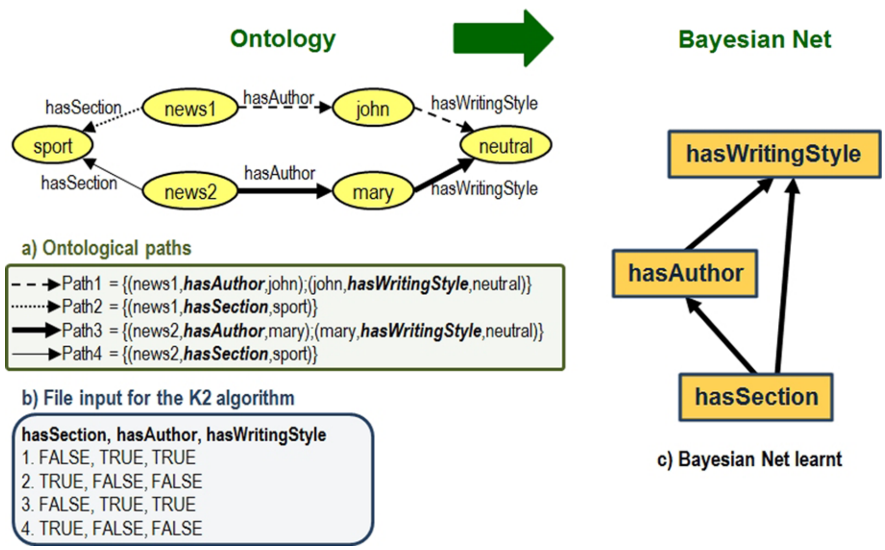


Fig. 3 A Bayesian Net example learnt from ontological paths

the first line of the "file input" there are all the possible ontological relations (in this example: *hasSection*, *hasAuthor*, *hasWritingStyle*) separated by commas. Then, each line represents a different ontological path (4 paths in this case). In the file, each ontological path is represented as an m-tuple where the variable *Xi* is *TRUE* if that relation belongs to the path, or *FALSE* otherwise. For example, the ontological path *Path1* = {(news1, ***hasAuthor***, john); (john, ***hasWritingStyle***, neutral)} contains only the second and third relations (but not the first one which is *hasSection*); thus, it is represented as (FALSE, TRUE, TRUE). Later, we use the K2 algorithm [21] to learn the Bayesian Net depicted in Fig. 3(c). As we create one Bayesian Net for each user action, from each net we learn the probability of each ontological relation type of being relevant to a certain user action.

Step 3: Ontology segment extraction

We want to learn the context that is relevant to the user to execute a certain action. Therefore, we should learn from all the context attributes which ones influence the user's actions. In this proposal, context attributes are represented as ontology concepts. Then, the objective is to find the ontology segments that represent each context.

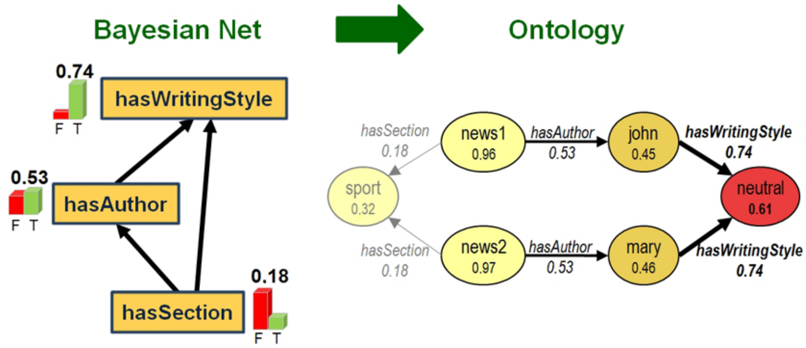
As we group the association rules according to the same user action, from each group we can extract the context attributes that are relevant to certain user action. As we mentioned above, the rule consequent is a user action and the rule antecedent contains the context attributes. Therefore, the context attributes of the rule antecedent are the starting point of the ontology segment extraction. We use spreading activation techniques [21, 23] to extract the ontology segments that are relevant to the user. A spreading activation technique searches for related concepts in a semantic net (an ontology in our case) starting from an initial set of concepts.

We use the context attributes discovered by the association rules (in Step 1) as the initial concepts of our spreading activation algorithm. Initially, the extraction process starts by labeling a set of concepts with weights that we called *relevance levels* and we denoted as *Rel* in Fig. 4. Then, the process iteratively propagates that relevance to other concepts following the ontological relations. The *relevance levels* are calculated from the confidence level of the association rules. Each concept accumulates the relevance received from all its neighbors. Following the previous example, "neutral" is influenced by "john" and "mary".

The relevance levels decay as activation propagates through the ontology. This is due to a weight (denoted as *propagation level*) associated with the ontological relations. We use the probabilities learnt by the Bayesian Nets (in Step 2) to label the ontological relations. The *propagation level* of a relation determines the percentage of the relevance of the source concept that is propagated to the target concept. For instance, a relation with a *propagation level* of zero implies that no relevance is propagated through that relation. As shown in Fig. 4, the relevance of "neutral" in a certain time t is calculated as follows (where $\beta = 0{,}9$ is a decay factor):

$$\begin{aligned}
Rel_t(\text{neutral}) &= Rel_{t-1}(\text{neutral}) + [\mathit{Influence}(\text{john}) \\
&\quad + \mathit{Influence}(\text{mary})] \\
&= Rel_{t-1}(\text{neutral}) + [(Rel_t(\text{john}) \\
&\quad \times \mathit{Prop}(\text{hasWritingStyle}) \times \beta) \\
&\quad + (Rel_t(\text{mary}) \\
&\quad \times \mathit{Prop}(\text{hasWritingStyle}) \times \beta)] \\
&= 0 + [(0{,}45 \times 0{,}74 \times 0{,}9) \\
&\quad + (0{,}46 \times 0{,}74 \times 0{,}9)] \\
&= 0{,}61
\end{aligned}$$

Fig. 4 Ontology segment extraction example



When spreading the relevance levels through the ontology, we find those concepts into which different concepts converge ("neutral" in the example). In this way, we discover those concepts that are contextually related to the initial ones. In the example, we discover that the user read the "news" because they were written by "journalists" with a "neutral" writing style, and not because they belong to the "sports" section. Finally, the ontology segments are composed of those concepts whose *relevance level* is beyond a predefined threshold. The ontology segments obtained compose the user profile.

6 Case study: user profiles in SAVER

In this section we present a case study in the domain of e-learning in which our approach was evaluated. In particular, we show how to learn the relevant context for each task executed by a student and how to gather it in the User Profile proposed herein.

E-learning environments have gained a wide acceptance and are currently being used by a wide variety of students with different skills, backgrounds, interaction preferences, and learning styles. Such diversity has brought new opportunities and new challenges [24]. As e-learning systems evolve toward more complex configurations, modeling context becomes more important either to personalize or to reuse that contextual information [3]. Therefore, it is important to discuss ways to improve these environments by considering the students' context. This is the reason why we decided to test our approach in this domain. Our intention is to increase even more the personalization capacities of actual e-learning systems by making use of ontologies to model the context of students in different scenarios. Also, we aim to test our proposal's ability to learn the relevant context for the student in concrete e-learning situations.

SAVER[1] is an e-learning system developed at UNICEN University. SAVER observes the student's behavior while he is taking a web course. The system stores the student's actions in a database together with all the information that characterizes the e-learning situation. Thus, we generate user's situations starting from all the available information in the database. A user's situation is a tuple (C, t) where t is a task that the student executed in SAVER (such as submit an evaluation or read a topic explanation) and C is a group of contextual attributes in the form $C = \{a1, \dots, an\}$ that describes the situation in which that task was executed. These situations are the input of our learning algorithm (Fig. 2). Figure 5 shows a partial view of the ontology used in this case study.

7 Experimental results

In this section, we present the results of our experiments aiming to test the context-aware user profiling technique proposed. We carried out two different experiments to validate our user profile learning. First, we compared the performance of our user profiling technique with and without the use of ontologies. Second, we analyzed our technique's ability to detect the contextual preference changes of the user and to update the ontology segments in order to reflect those changes.

7.1 Experimental data

We evaluated our approach with a set of 52 System Engineering students. These students took a course on Artificial Intelligence using SAVER. Starting from the records of the students' interactions gathered in the database of SAVER, we built a user profile for each of them. Two types of user profiles were implemented: (i) a semantically enriched user profile using the proposed approach; and (ii) a user profile using only association rules. The objective was to compare both profiles so as to show the advantages and disadvantages of our proposal.

In total, 1877 user situations from 52 different students were gathered in SAVER database. The situations were randomly divided in training and testing situations. Two thirds

[1] http://www.e-unicen.edu.ar/.

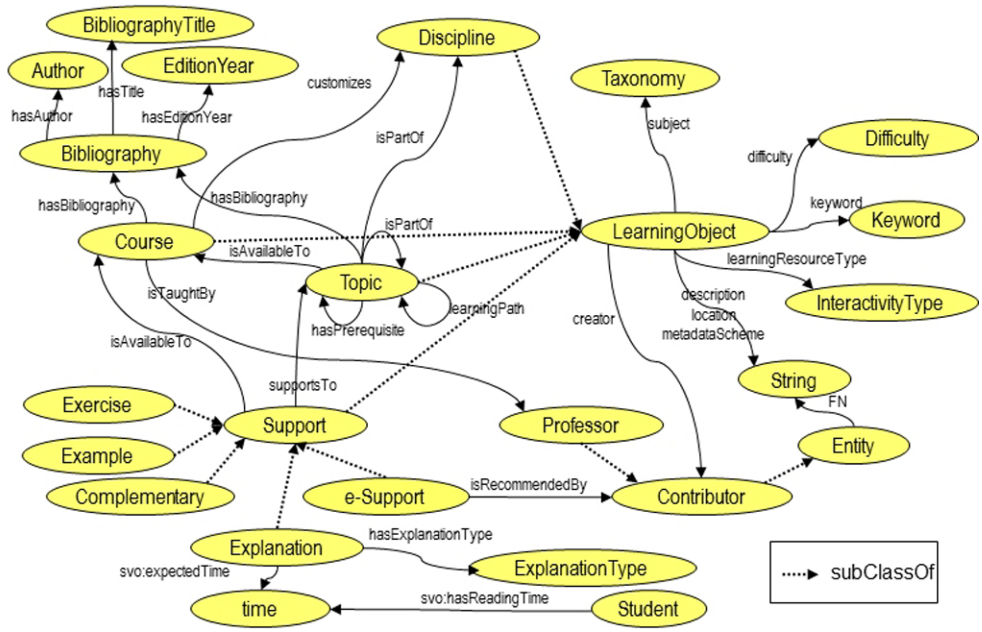


Fig. 5 Ontology partial view

of the situations (1252) were used for training and the remaining (625) for testing. Some numeric and temporal variables were averaged (such as the beginning and ending dates of sessions). Each user situation was composed of the explanation that the student read and contextual attributes that described the situation in which that reading took place. Then, we looked in the ontology for the common semantics to those attributes to find the relevant context for that student using only the training situations. Finally, we validated the learnt context using the test situations. In the test situations, there were explanations that the student considered interesting (because he/she has read them) but they were not considered in the learning. Therefore, we measured if starting from the learnt context we were able to detect the interesting explanations in the test situations.

7.2 First experiment

In the first experiment, we compared the performance of explanation retrieval: (a) with and (b) without the use of our semantically-enriched user profile. For this purpose, we implemented an experimental personal agent in order to test our user profile techniques. The personal agent suggested interesting explanations to the student first by (a) using a rule-based user profile and then (b) using our semantically-enriched user profile. Consequently, we first built an individual user profile for each student involved in the experiment using only association rules. Then, we built a second context-aware user profile enriching those association rules with ontological knowledge as described in the previous sections. To generate association rules, we used WEKA [25], which is a well-known data mining tool. We empirically determined the parameters used to run the algorithms (*confidence* and *support* [18]): minconf $= 0.8$, minsup $= 1/N$, where N is the number of instances in data.

Our goal is to retrieve interesting explanations for students. Therefore, *precision* and *recall* are two possible measures of effectiveness. The metrics are defined as follows:

$$\text{Precision} = \frac{\text{Number of interesting explanations retrieved}}{\text{Number of retrieved explanations}}$$

$$\text{Recall} = \frac{\text{Number of interesting explanations retrieved}}{\text{Number of interesting explanations}}$$

With *precision* we measured if *only* interesting explanations are retrieved; whereas with *recall* we measured if *all* interesting explanations are retrieved. We inferred that a student was interested in an explanation when he/she read it. All the explanations that a certain student read were gathered in SAVER database. We first calculated precision and recall by using the rule-based user profile for each student in the experiment. Then, we recalculated both metrics but by using the semantically-enriched user profile.

We created separate precision and recall graphs to facilitate the comparison and contrast between the rule-based user profile and the semantically-enriched one. Figure 6 presents the experiment results. Figure 6(a) shows the precision values, whereas Fig. 6(b) shows the recall values for both approaches. To make the results comparable, the values shown were obtained by averaging the precision and recall for the different data sets belonging to the students.

The experimental results show improvements in the recommendation of explanations to students with the use of our approach. As visualized in Fig. 6(a), the recommendations using our profile exceeds the recommendations with

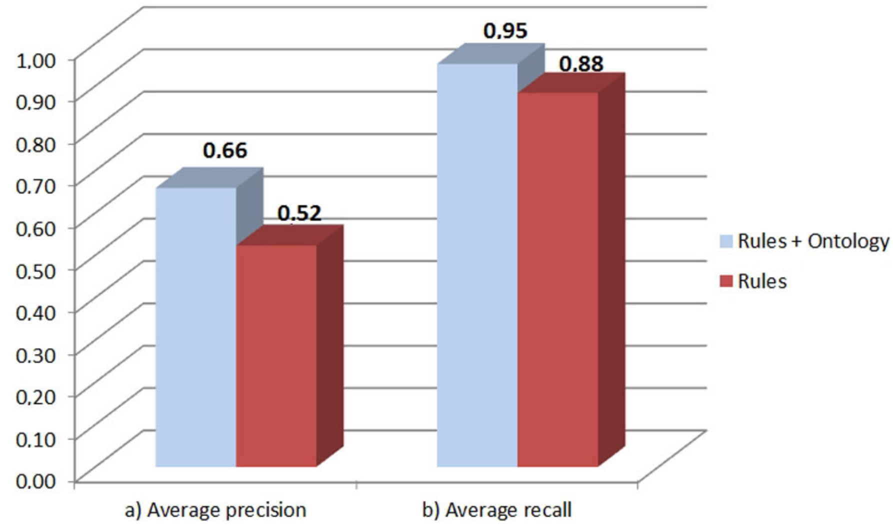


Fig. 6 Comparative performance of user profile techniques with and without semantics: (**a**) average precision and (**b**) average recall

rules from 0,66 to 0,52 in average precision. Therefore, our proposal has an improvement of 14% in average precision. On average, without using our user profile, only around 50% of interesting explanations were retrieved. Thus, we can affirm that the semantically-enriched user profile outperforms the rule-based one on average precision. Additionally, recall was also improved, as observed in Fig. 6(b). Our semantically-enriched user profile had a higher recall than the rule-based one. The recommendations of our user profile have greater recall (0,95) in comparison with the recommendations with rules (0,88). However, the average recall values of both proposals are very close. Our approach produced an average recall increment of 0,06.

Some conclusions can be drawn from this experiment. In general, the results show that the recommendations of explanations using personalized user profiles achieve a good performance in both precision and recall. However, the improvements made were not as significant as we expected. We have two possible explanations for these results. First, the reduced size of the ontology used did not allow us to detect useful semantic relationships among the attributes. Second, the data corpus was not large enough for bringing representative examples of students' behaviors.

7.3 Second experiment

In the second experiment, we tested how our approach reacts to changes in students' contextual preferences. Initially, we imagined a scenario in which the student prefers a certain context, and then we considered a different scenario in which the student completely changes his/her contextual preferences. The objective of this experiment is to analyze whether the proposed approach can detect these changes and how the Ontology Segment is updated so as to incorporate those changes.

Considering the ontology described in Fig. 7,[2] suppose that student Peter initially prefers to read explanations of the topic "association rules" (denoted as "*Assoc-Rules*" in Fig. 7) with a "*Low*" difficulty level (scenario 1). Later, when he gets more training in the use of the e-learning system and learns a bit more about it he changes his behavior and starts reading explanations about the topic "Bayesian networks" (denoted as "*Bayes*" in Fig. 7) with "*Medium*" difficulty level (scenario 2). Figure 8(a) shows the changes in the relevant levels of the concepts in the ontology. This figure depicts the evolution of the relevant levels as the student changes his contextual preferences. Y axis describes the relevance levels, whereas X axis depicts the executions of our algorithm.

Figure 8(a) shows how relevance levels are updated through both scenarios. Initially, we executed our algorithm for scenario 1 considering that there is no previous ontology segment. Therefore, the algorithm quickly learns the relevant context for that scenario. This can be seen in executions 1 and 2, where the relevance levels of the relevant concepts ("*Assoc-Rules*" and "*Low*") increase in a considerable way. Then, after the second execution, we began to incorporate evidence from scenario 2. As a consequence, the relevance levels of the concepts in the first scenario began to gradually decrease, whereas the relevance of the concepts of the second one increased. First, we noted that the relevance values of "*Bayes*" remained almost constant at 1. The reason behind this is that one of the relevant concepts of the first scenario ("*Probability*") is a topic related to "*Bayes*" as well as "*Assoc-Rules*". Therefore, the evidence of the first scenario for "*Bayes*" was not easily "forgotten", and made that "*Bayes*" remain with a high relevance. Second,

[2] In Fig. 7, concepts are represented as circles while instances are represented as squares.

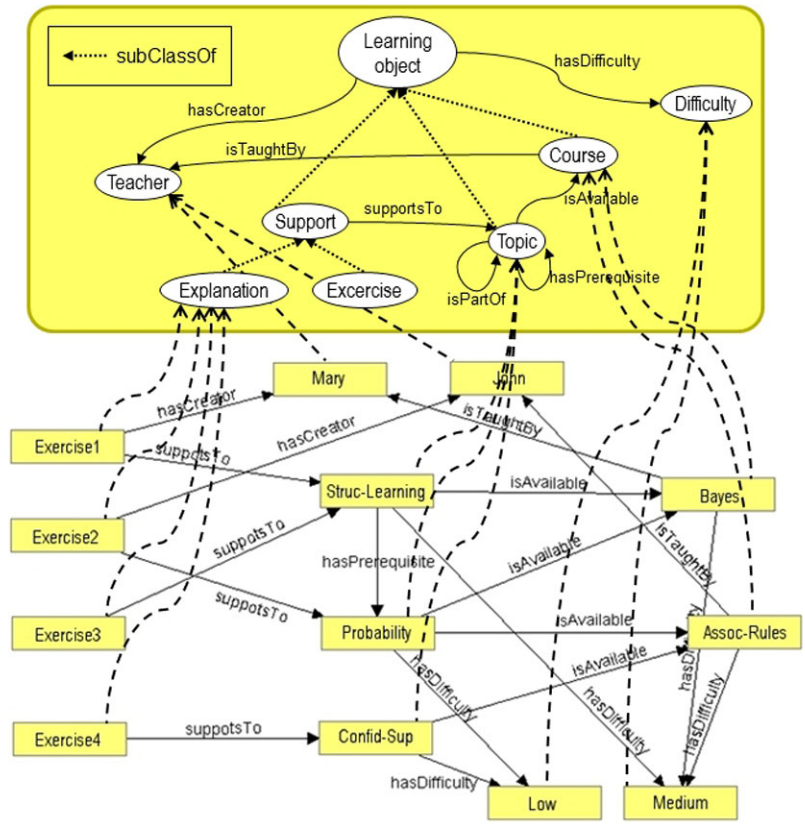


Fig. 7 Ontology example used in the second experiment

"*Struc-Learning*" and "*Medium*" increase because they are two concepts directly related to "*Bayes*". However, through the subsequent algorithm's executions "*Media*" decreased a bit because it is related to "*Assoc-Rules*" (that decreased in scenario 2). Finally, we see a decrease in the relevance of concepts: "*Confid-Sup*", "*Low*", "*Assoc-Rules*" and "*Probability*" which were relevant in the first scenario but not in the second.

Another interesting aspect in this experiment is to analyze changes in the propagation levels of the ontological relations. Remember that the propagation levels are calculated using Bayesian nets. Therefore, each time we get evidence of user behavior changes, such nets are updated accordingly. When entering new evidence in a Bayesian net, all its probabilities are recalculated. Figure 8(b) shows the propagation levels in scenario 1 and scenario 2 respectively. Analyzing this figure, we can see that all propagation levels change between scenarios. The most important change is in the *hasPrerequisite* relation: it has a value of zero in the first scenario and 0,58 in the second one. The reason for this change is that there is no ontological path that starts in a relevant concept of the first scenario and passes through that relation. In other words, scenario 1 was about "association rules" and the topics of that subject have no prerequisites. However, in the second scenario, the subjects do have prerequisites; thus, the relation *hasPrerequisite* has a value of 0,58.

In summary, as shown in this experiment, the proposed approach is able to adapt the ontology segment according to the user's behavior changes. At this point, it is important to note that variations in the relevance levels do not change abruptly because of the "memory" of previous preferences in the model. The gradual decline in the relevance levels is used to protect the original relevance levels of the segment and to avoid drastic changes. By using this strategy,the ontology segments will immediately reflect any change in the user's behavior pattern. If the change becomes a new behavior pattern, it will be emphasized; if it is merely sporadic or erratic, however, the segments will not be considerably affected.

8 Conclusions

In this paper, we propose a novel approach for context-enriched user profiles based on the integration of data mining and ontologies. Particularly, we are interested in knowing to what extent data mining techniques can be used for user profile generation, and how to utilize ontologies for user profile improvement. Consequently, we semantically enriched a user profile using association rules, Bayesian networks and ontological knowledge in order to improve an agent's performance. Herein, the user profile is enhanced by enriching it with the semantics of an ontology. Ontologies,

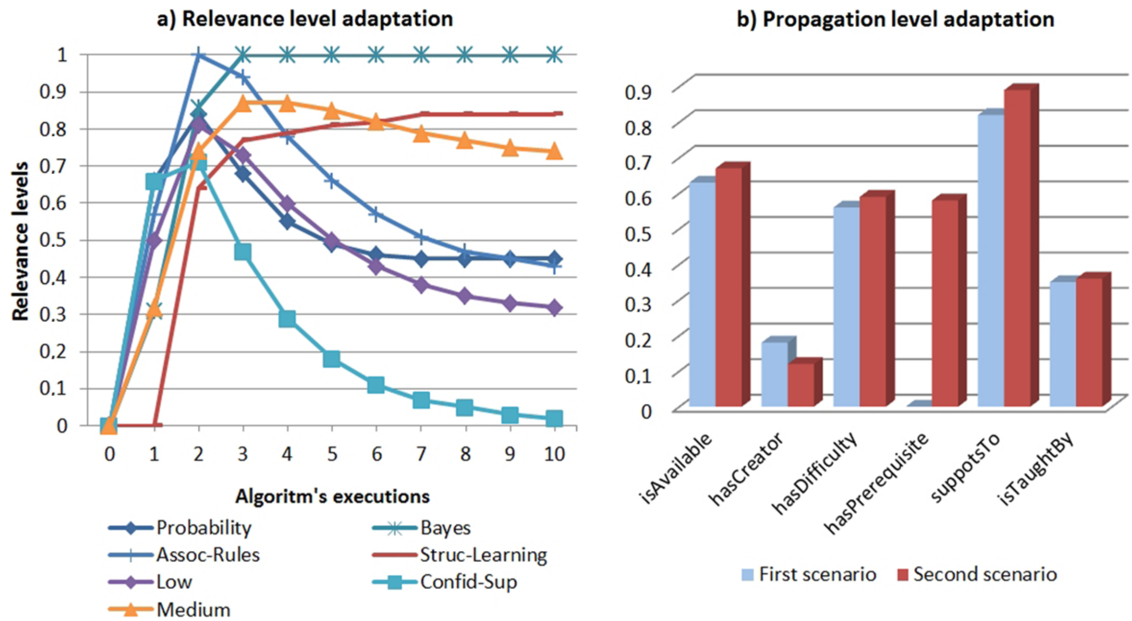


Fig. 8 (**a**) Relevance level adaptation and (**b**) propagation level adaptation to changing user contextual preferences

with their supporting technologies, are a useful knowledge representation for disambiguating heterogeneous user information.

Our innovative approach combines the implicit relevant context learned at runtime, with a persistent, more general ontology segment representation of user profiles. We learn the relevant context to the user on the basis of the user's behavior observation. We use ontology segments to represent relevant contexts in the user profiles. To build ontology segments, we first generated association rules from simple context attributes. In our approach, we mapped context attributes as ontological concepts, and context attribute values as instances. Then, we compared the context attributes obtained from the association rules to find out if they have similar semantics. When discovering attributes with similar semantics, these attributes are compressed into their shared concept. In particular, we searched for the common relations among context concepts in the domain ontology. Once we discovered the shared concept of context attributes, we extracted the ontology segments that represent the relevant context to the user. Finally, the relevant ontology segments were attached to user's preferences and to certain parameters, such as certainty and confidence. This approach did not completely remove the uncertainty of predicting implicit user's preferences, but it could significantly improve agent precision in a considerable number of cases.

The encouraging experimental results show the usefulness of including semantics in a user profile as well as the advantages of integrating agents and data mining using ontologies. Although conventional data mining techniques can help agents, more efficient data mining techniques are needed to mine the data gathered from the user's behavior into user profiles. Consequently, we propose a novel ontology-based agent mining integration. In this work, we took advantage of data mining techniques enriched with the semantics gathered within an ontology to produce highly relevant user profiles. Our ontology segments range from only a few interrelated concepts to complex ontologies describing a complete domain. The ontology segments are a powerful approach to represent different types of relations among context attributes, which also benefit from existing ontological technologies. As a conclusion of this work, we believe that the proposed integration is significant since it can improve the effectiveness of personal agents.

References

1. Schiaffino S, Amandi A (2006) Polite personal agents. IEEE Intell Syst 21(1):12–19
2. Godoy D, Amandi A (2005) User profiling for web page filtering. IEEE Internet Comput 9(4):56–64
3. Eyharabide V, Gasparini I, Schiaffino S, Pimenta M, Amandi A (2009) Personalized e-learning environments: considering students' contexts. In: Proceedings of WCCE 2009, world conference on computers in education IFIP—international federation for information processing, pp 48–57
4. Kim H-R, Chan P (2008) Learning implicit user interest hierarchy for context in personalization. Appl Intell 28:153–166
5. Gruber T (1993) A translation approach to portable ontology specifications. Knowl Acquis 5:199–220
6. Tao X, Li Y, Zhong N (2010) A knowledge-based model using ontologies for personalized web information gathering. Web Intell Agent Syst 8:235–254
7. Duong T, Uddin M, Li D, Jo G (2009) A collaborative ontology-based user profiles system. In: Proceedings of the 1st interna-

tional conference on computational collective intelligence, semantic web, social networks and multiagent systems, pp 540–552
8. Zhou X, Wu S-T, Li Y, Xu Y, Lau R, Bruza P (2006) Utilizing search intent in topic ontology-based user profile for web mining. In: Proceedings of the 2006 IEEE/WIC/ACM international conference on web intelligence, pp 558–564
9. Sutterer M, Droegehorn O, David K (2008) User profile selection by means of ontology reasoning. In: Proceedings of the 2008 fourth advanced international conference on telecommunications, pp 299–304
10. Mylonas P, Vallet D, Castells P, Fernandez M, Avrithis Y (2008) Personalized information retrieval based on context and ontological knowledge. Knowl Eng Rev 23:73–100
11. Gauch S, Chaffee J, Pretschner A (2003) Ontology-based personalized search and browsing. Web Intell Agent Syst 1:219–234
12. Tao X, Li Y, Zhong N (2010). A personalized ontology model for web information gathering. IEEE Trans Knowl Data Eng 99 (PrePrints)
13. Sieg A, Mobasher B, Burke R (2007) Representing context in web search with ontological user profiles. Model Using Context 4635:439–452
14. Lee Y-S, Cho S-B (2011) Exploiting mobile contexts for Petri-net to generate a story in cartoons. Appl Intell 34:1–18
15. Eyharabide V, Amandi A (2007) An ontology-driven conceptual model of user profiles. In: Proceedings of ASAI 2007, 9th Argentine symposium on artificial intelligence, August 27–28, Mar del Plata, Argentina, pp 101–115
16. Seidenberg J, Rector A (2006) Web ontology segmentation: analysis, classification and use. In: WWW '06: proceedings of the 15th international conference on world wide web. ACM Press, New York, pp 13–22
17. Agrawal R, Shafer J (1996) Parallel mining of association rules. IEEE Trans Knowl Data Eng 8(6):962–969
18. Shah D, Lakshmanan L, Ramamritham K, Sudarshan S (1999) Interestingness and pruning of mined patterns. In: ACM SIGMOD workshop on research issues in data mining and knowledge discovery
19. Eyharabide V, Amandi A (2008) Semantic spam filtering from personalized ontologies. J Web Eng, 7(2):158–176
20. Yap G-E, Tan A-H, Pang H-H (2008) Explaining inferences in Bayesian networks. Appl Intell 29:263–278
21. Cooper G, Herskovits E (1992) A Bayesian method for the induction of probabilistic networks from data. Mach Learn 9(4):309–347
22. Dechter R (1998) Bucket elimination: a unifying framework for probabilistic inference. In: Proceedings of the NATO advanced study institute on learning in graphical models, pp 75–104
23. Crestani F (1997) Application of spreading activation techniques in information retrieval. Artif Intell Rev 11(6):453–482
24. Iglesias A, Martinez P, Aler R, Fernandez F (2009) Learning teaching strategies in an adaptive and intelligent educational system through reinforcement learning. Appl Intell 31:89–106
25. Witten I, Frank E (2005) Data mining: practical machine learning tools and techniques