Semantically Enhanced Recommender Systems

Manuela Ruiz-Montiel and José F. Aldana-Montes

Departamento de Lenguajes y Ciencias de la Computación,
University of Málaga, España
{mruizmonti,jfam}@lcc.uma.es

Abstract. Recommender Systems have become a significant area in the context of web personalization, given the large amount of available data. Ontologies can be widely taken advantage of in recommender systems, since they provide a means of classifying and discovering of new information about the items to recommend, about user profiles and even about their context. We have developed a semantically enhanced recommender system based on this kind of ontologies. In this paper we present a description of the proposed system.

Keywords: Ontologies, recommender systems, user modelling, personalisation.

1 Introduction

Recommender Systems aim to filter potentially useful items from a vast amount of data. We propose a family of Semantic Recommender Systems that, by means of domain ontologies describing both the users and the items, will enhance traditional approaches. Ontological reasoning allows the discovery of semantic features that play a very important role in the recommendation process, since they are the basic principles underlying the users' choices. Moreover, the use of ontologies will help us to address some of the typical problems of recommender systems.

2 Semantic Recommender Systems

Traditional Collaborative Filtering algorithms proceed by calculating similarities between users or between items [1]. These similarities are based on the ratings the users have given to the items. In the first case (user based), a user will receive recommendations made up of the items that similar users liked best. In the second case (item based), the recommended items will be those that are similar to the ones that the user loved in the past. This latter approach is known to be more efficient, since the similarities can be calculated offline [2]. This type of recommender system has several problems, like the *new user* problem, the *new item* problem and the *gray sheep* problem [1]. The *new user* or *item* problem arise when a new user or item is stored in the system. A new user will not

R. Meersman, P. Herrero, and T. Dillon (Eds.): OTM 2009 Workshops, LNCS 5872, pp. 604–609, 2009.

have made any previous evaluations about the items, so it will be difficult to compute his/her similarities with respect to other users; and a new item which has no ratings will have the same problem. The *gray sheep* problem arises when nobody likes an item, so it will not be recommended, though it is possible that some users may like it. If semantic features are taken into account, then the similarities could be computed according to them. Indeed, the semantic features are the underlying reasons for the items or users to be similar or not. Moreover, the problems that arise when we compute the similarities in a collaborative way, now disappear.

On the other hand, Content Based Recommender Systems make recommendations considering the items' and users' semantic features[1]. The items recommended to a given user will be those whose features match a computed user profile. These recommender systems are a perfect area in which to include ontologies, since they are based on the semantics.

Our approach for developing a new family of recommender systems is based on domain ontologies containing the semantic attributes both for items and users. We use OWL ontologies and a reasoner able to classify the described resources.

2.1 Related Work

Middleton, De Roure and Shadbolt have developed a recommender system based on an ontology which constitutes an is-a hierarchy of paper topics, in order to infer more accurate user profiles according to the categories within the taxonomy [3]. Mobasher, Jin and Zhou have taken advantage of an item ontology, which is part of the schema for a relational database, in order to compute similarity correlations between items [2]. Wang and Kong also use an ontology to calculate the correlations between items, and they use a very similar algorithm to the one we present in this paper; but they do not use the ontology to infer the semantic features, since they explicitly specify them [4]. In [5] and [6], the authors also propose the use of semantic descriptions of items and user profiles. In [7], the authors demonstrate that taking into account the underlying semantics in recommender systems leads to better results. We propose the use of semantically rich ontologies (and not only hierarchies) in order to automatically infer implicit information of the items to recommend and the users looking for items, so we can take advantage of it and make more accurate recommendations. Our ontologies have been made manually, gathering information from the tourism domain.

2.2 Domain Ontologies

We have developed two main ontologies. The first one, called the **Item Ontology**, describes the items according to several criteria depending on the specific domain. The second one is the **User Ontology**, which classifies the users according to personal data provided such as gender, age or occupation. This ontology will infer new users' preferences with respect to their personal data. In our case (tourist services), the preferences will be related to different areas such as price, situation or quality.

The Item Ontology: An Example. We have developed an ontology in the tourism domain, classifying the items in categories such *Inexpensive Service*, *High Quality Service*, and many other more traditional ones such as *Restauration Service* or *Accommodation Service*. The hierarchy does not end at this first level, but it still continues refining other aspects of each category. E.g., inside 'Accommodation Service' we can find classes like *Charming Accommodation Service*. Each point of interest described in the ontology will be member of a set of categories, which will mould the semantic profile of the item.

Let us consider a point of interest described in the Item Ontology. Initially, the item only belongs to the class *Tourist Service*, but two basic features have been asserted: the item has an average price of 20 euros for the set menu and its speciality is the lobster. The ontology will infer that this point of interest belongs to several categories: *Restauration Service*, *Expensive Service* and *Savoury Food Service*. These categories mould the *semantic profile* of the item.

The User Ontology: An Example. The User Ontology is designed depending on the item domain. Nevertheless, the user ontologies for different item domains will have several features in common, i.e., personal data properties: gender, age, interests, etc. The diference between the ontologies will reside in the preferences inferred. The User Ontology will import the terms defined in the Item Ontology, since the user and the item profiles must be calculated with respect to exactly the same domain. In our tourist example, we have defined some classes describing users and their respective preference assumptions. For example, the users in the *Young User* category, will prefer *Young-oriented* tourist services.

2.3 Semantic Filtering Recommendation

In Collaborative Filtering Systems, the recommendation process is based on similarity calculation between users or items. The first approaches were implemented with user-user similarity, computed by correlations [1] over the rating vectors. The principal shortcoming of this philosophy is the fact that these vectors are not static: they change continuously and the similarities cannot be computed offline. The recommendation process is neither efficient nor scalable, since the complexity is exponential with respect to the number of users. On the other hand, approaches based on item-item similarities [8,9] have turned out to be more efficient and scalable. The similarities of items are naturally more static than similarities between users, so their computation can be done offline [2]. A great advantage is the fact that now we can take other sources of information into account, different from the rating vectors, i.e., models for the items that include their semantic features.

Recommendation Process. Let us imagine there is a user in the system who has rated several items. Some of the ratings will be positive and others will be negative. If we take into account the positive ones, we can gather a set of well-rated items from which to calculate neighborhoods. Given a well-rated item, its neighborhood is the set of the n most similar items in the system. The similarity between two items is calculated as follows:

$$sim_{i,j} = \frac{|SIP(i) \bigcap SIP(j)|}{max(|SIP(i)|, |SIP(j)|)}$$

Where *SIP*(i) is the *Semantic Item Profile* of the item i, calculated by means of the Item Ontology -i.e, it is the set of semantic categories the item i belongs to. Note that similarities range from 0 to 1. Once we have computed all the neighborhoods of the well-rated items, we recommend those items in the union of all the neighborhoods that satisfy the next two conditions: the item has not been used by the selected user and the *Recommendation Factor* is bigger than a certain number, called *Recommendation Threshold*, which is a measure of how good the recommendation will be for an user. It is calculated as follows:

$$RF(i) = r(father) * sim_{i,father}$$

Where *father* is the item from which the neighborhood was calculated. If an item belongs to more than one neighborhood, then we take into account the biggest factor of all the possible recommendations. The *Recommendation Threshold* that we use to filter the items depends on the ratings domain. If the maximum rating is, e.g., 5 points, then we can consider a threshold of 3. This implies that the similaritiy of the recommended item with respect to its father must be bigger than 3/5 in order to be recommended. If the father has a rating of 4 points (out of 5), then the similarity must obviously be greater than 4/5: if the father has a lower rating, then the similarity has to be stronger in order to make the item a good recommendation. Nevertheless, this number may be parametrized depending on the number of recommendations we want to compute. Even if we do not consider any threshold, the items can be ordered according to the *Recommendation Factor* and we can take the first n items of this list.

Using this method we avoid the *new item* problem, since we do not need the ratings of the recommended items; as well as the *gray sheep* problem, because we focus on all the posible items in the system, computing their similarities. Nevertheless, the *new user* problem still remains: if the user has not given any rating, then the neighborhoods cannot be calculated.

2.4 Content Based Semantic Recommendation

Another family of Recommendation Systems are Content Based approaches. They share in common a means for describing both the items and the users, and a technique for comparing the items to the user profile [1]. Obviously, our means for describing both the items and the users will be the domain ontologies described in previous sections.

The user profile is a combination of two sources of information: the User Ontology and the record of ratings the user has given to the used items. Through the User Ontology we can extract a semantic profile in a similar way to how we computed the *Semantic Item Profile*. So this profile is a set of categories the user belongs to, given his/her personal data such age, occupation, interest, etc. But we have another very powerful source of information: the ratings the user has given to the items he or she has used. We can aggregate the item profiles of the well-rated items, so we can mould a map of semantic categories, each one of them accompanied by a number indicating its importance. This importance will grow if the category refers to more than one well-rated item, and it will also rise if the well-rated items have better ratings. If we mix this map with the semantic profile extracted from the User Ontology, we can consider the result as an accurate *Semantic User Profile* (*SUP*) of the selected user.

The recommendation process. Now we have computed the *SUP*, we can recommend items to the user. The items will be recommended if they fulfill both following conditions: the item has not been used by the selected user and the *Semantic Correspondence* between the *SIP* (*Semantic Item Profile*) and the *SUP* is bigger than a certain number, called *Recommendation Threshold*. The *Semantic Correspondence* (*SC*) is calculated as a ponderate product of the *SIP* and the *SUP*. The pseudocode is as follows:

```
SC := 0
for (profileElement in SIP){
  if (profileElement belongs to SUP){
    SC := SC + importance(profileElement, SUP)
  }
}
```

The *Recommendation Threshold* is a number that depends on the importance factors that we gave to the different elements of the *Semantic User Profile*, and also depends on the average numbers of profile elements in the *Semantic Item Profiles*. If in these profiles there is an average number of I elements, and the importance factors move from 0 to 100, then (I/2)*75 (half the features with a good importance) could be a good threshold in order to recommend an item or not. Nevertheless, this number may be parametrized. Even if we do not consider any threshold, the items can be ordered according to the *Semantic Correspondence*.

Using this method we avoid the *new user* problem, since we can obtain a simple user profile from his or her personal data. Nevertheless, the more items the user rates, the more accurate the computed profile is. With respect to the *new item* and *gray sheep* problems, both are addressed since we focus on all the items of the system, computing their *Semantic Correspondence* with the *Semantic User Profile*.

3 Conclusions and Future Work

Semantics and ontologies are a powerful tool for improving the performance of recommender systems, since they allow us to avoid some of the typical problems normally involved in this kind of process. Moreover, when we talk about similarities or correspondence between items or item-user, we are really referring to the semantics beneath the users and the items. We have developed two Semantically Enhanced Recommender Systems based on two different ontologies: the Item Ontology and the User Ontology, in order to discover user and item profiles.

A Context Ontology could be included in this kind of recommender systems, since contextual information is very powerful in some situations in order to filter the available items. Moreover, an evaluation of these semantic enhanced algorithms with respect to traditional approaches needs to be done, in terms of precision and recall and some other semantic-specific criteria.

Acknowledgments. This work was supported by the ICARIA Project Grant (Excellence Project, Innovation, Science and Enterprise Ministry of the regional government of the *Junta de Andalucía*), TIN2008-04844 (Spanish Ministry of Education and Science), *Proyecto Piloto Formación y desarrollo de tecnología aplicada a la biología de sistemas*, P07-TIC-02978 (Innovation, Science and Enterprise Ministry of the regional government of the *Junta de Andalucía*) and the project *Creación de un soporte tecnológico para un sistema de información turística*, FFI 055/2008 (Tourism, Sport and Commerce Ministry of the regional government of the *Junta de Andalucía*).

References

1. Adomavicius, G., Tuzhilin, A.: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Trans. Knowl. Data Eng. 17(6), 734–749 (2005)
2. Mobasher, B., Jin, X., Zhou, Y.: Semantically Enhanced Collaborative Filtering on the Web. In: Berendt, B., Hotho, A., Mladenič, D., van Someren, M., Spiliopoulou, M., Stumme, G. (eds.) EWMF 2003. LNCS (LNAI), vol. 3209, pp. 57–76. Springer, Heidelberg (2004)
3. Middleton, S.E., Shadbolt, N., De Roure, D.: Ontological user profiling in recommender systems. ACM Trans. Inf. Syst. 22(1), 54–88 (2004)
4. Wang, R.-Q., Kong, F.-S.: Semantic-Enhanced Personalized Recommender System. In: International Conference on Machine Learning and Cybernetics, vol. 7, pp. 4069–4074 (2007)
5. Liu, P., Nie, G., Chen, D.: Exploiting Semantic Descriptions of Products and User Profiles for Recommender Systems. Computational Intelligence and Data Mining, 179–185 (2007)
6. Ziegler, C., Schmidt-Thieme, L., Lausen, G.: Exploiting semantic product descriptions for recommender systems. In: Proc. 2nd ACMSIGIR SemanticWeb and IR WS (2004)
7. Moshfeghi, Y., Agarwal, D., Piwowarski, B., Jose, J.M.: Movie Recommender: Semantically Enriched Unified Relevance Model for Rating Prediction in Collaborative Filtering. In: Boughanem, M., et al. (eds.) ECIR 2009. LNCS, vol. 5478, pp. 54–65. Springer, Heidelberg (2009)
8. Linden, G., Smith, B., York, J.: Industry Report: Amazon.com Recommendations: Item-to-Item Collaborative Filtering. IEEE Distributed Systems Online 4(1) (2003)
9. Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. ACM Trans. Inf. Sys. 22(1), 143–177 (2004)