

Extracting Multilayered Semantic Communities of Interest from Ontology-based User Profiles: Application to Group Modelling and Hybrid Recommendations

Iván Cantador, Pablo Castells

*Departamento de Ingeniería Informática, Escuela Politécnica Superior,
Universidad Autónoma de Madrid, 28049, Madrid, Spain
{ivan.cantador, pablo.castells}@uam.es*

Abstract

A Community of Interest (CoI) is formed by a group of individuals who share and exchange ideas and thoughts about a given interest or passion. However, they are often not aware of their membership to the community, and they may know or care little about each other outside of this clique. This paper describes a proposal to automatically identify CoI from the tastes and preferences expressed by users in personal ontology-based profiles. The proposed strategy clusters those semantic profile components shared by the users, and according to the clusters found, several layers of interest networks are built. The social relations of these networks might then be used for different purposes. Specifically, we outline here how they can be used to model group profiles and make semantic content-based collaborative recommendations.

***Keywords:** community of interest, ontology, user profile, group modelling, content-based collaborative filtering*

1. Introduction

Communities of Practice (CoP) have been defined as groups of people who get involved in a collective work within a shared domain (Wenger, 1998). These people collaborate over a period of time, sharing ideas and experiences in order to find solutions for a particular practice. However, it is very often the case that the membership to a community is unknown. In many social applications, a person describes his interests and knowledge in a personal profile to find people with similar ones, but he is not aware of the existence of other (directly or indirectly) related interests and knowledge that might be useful to find those people. Further, depending on the context a user can be interested in different topics or groups of people. In both cases, a strategy to automatically identify CoP might be very beneficial (Alani, O'Hara & Shadbolt, 2002).

Communities of Interest (CoI), understood as a particular case of CoP, have been defined as groups of people who share a common interest or passion. In this paper, we

propose a novel approach towards building multilayered CoI by analysing the individual preferences described in ontology-based user profiles. Like in previous approaches (Liu, Maes, & Davenport, 2006), our method compares profiles of interests for semantic topics and concepts in order to find similarities among users. But in contrast to prior work, we divide the profiles into clusters of cohesive interests, and based on this, several layers of CoI are found. This provides a richer model of interpersonal links, which better represents the way people find common interests in real life.

Our approach is based on an ontological representation of the domain of discourse where user interests are defined (Castells, Fernández & Vallet, 2007). The ontological space takes the shape of a semantic network of interrelated concepts, and the profiles are described as weighted lists measuring the user's interests for those concepts. Taking advantage of the relations between concepts, we cluster the semantic space based on the correlation of concepts appearing in the preferences of individual users. After this, profiles are partitioned by projecting the concept clusters into the set of preferences of each user. Then, users can be compared on the basis of the resulting subsets of interests, in such a way that several, rather than just one, links can be found between two users.

The identified multilayered CoI are potentially useful for many purposes. For instance, in a recommendation environment there is an underlying need to distinguish different contexts within the user's preferences. Partitioned at various common semantic layers, the proposed CoI enable more accurate and context-sensitive recommendations. Thus, we shall present and evaluate several content-based collaborative filtering models which retrieve information items according to a number of real profiles and within different contexts. Further, our two-way space clustering offers a partition of the user space that can be exploited to build group profiles for sets of related users. These groups enable an efficient collaborative recommendation, by using the merged profiles as representatives of classes of users. To this end, we shall adapt several user profile merging techniques based on social choice theory (Masthoff, 2004).

The rest of the paper has the following structure. Section 2 summarises past works on CoP identification and social collaborative recommendations which are relevant for our proposal. Section 3 describes the ontology-based knowledge representation, upon which our personalised content retrieval processes described in section 4 are built. Our technique for building multilayered CoI is presented in Section 5. The exploitation of the derived CoI to enhance group modelling and content-based collaborative filtering is explained in Sections 6 and 7. Finally, some discussions are given in Section 8.

2. Related Work

In social systems, a profile of a user is mainly composed of his relationships with others, and possible additional information about these relationships: reliability, context, etc. Connected one to another, the users form graphs of social links, named as social networks. These links can be described explicitly or discovered directly from different sources of information, such as address books, IRC contact lists, or email boxes.

The relationships between users can also be formalised using ontologies. The Friend-Of-A-Friend (FOAF) ontology is one of the most popular in this area. It aims to create a network of machine-readable pages describing people, the links between them, and the things they do. Flink (Mika, 2005) is a system for the extraction, aggregation and visualisation of online social networks. It employs semantic technologies for reasoning with personal information extracted from a number of electronic information sources including web pages, emails, publication archives, and FOAF profiles. Extending the traditional bipartite model of ontologies (concepts and instances) with the social dimension leads to a tripartite model of the Semantic Web.

ONTOCOPI (Alani, O'Hara & Shadbolt, 2002) is another tool for discovering communities, by analysing ontologies of a given relevant domain. It attempts to disclose informal CoP relations by identifying patterns in the relations represented in ontologies, and traversing the ontology from instance to instance via selected relations. Performing experiments from an academic ontology, the authors show how the alteration of the weights applied to the ontology's relations affect the structure of the identified CoP.

Up to date, one of the most significant uses of social relations and CoP is the implementation of social collaborative recommendations. An ontological approach to user profiling within recommender systems is presented in (Middleton, Roure & Shadbolt, 2004). Working on recommending on-line academic research papers, the authors present two systems, Quickstep and Foxtrot, which create user profiles monitoring the behaviour of the users and gathering relevance feedback from them. The obtained profiles are represented in terms of a research paper topic ontology. Research papers are classified using ontological classes, and the recommender algorithms suggest documents seen by similar people on their current topics of interest.

In addition to explicit social relations, recent researches focus their attention in finding implicit relations among people, according to personal interests and preferences. Hence, (Liu, Maes, & Davenport, 2006) presents an implementation of 'taste fabrics', a

semantic mining approach to the modelling of personal tastes for different topics of interest. The taste fabric affords a flexible representation of user preferences, enabling a keyword-based profile to be ‘relaxed’ into a spreading activation on the taste fabric, and performing favourably in relation to classic collaborative filtering methods.

3. Ontology-based Knowledge Representation

In contrast to other approaches in personalised content retrieval, our approach makes use of explicit user profiles (as opposed to e.g. sets of preferred documents). Working within an ontology-based personalisation framework, user preferences are represented as vectors $\mathbf{u}_m = (u_{m,1}, u_{m,2}, \dots, u_{m,K})$ where $u_{m,k} \in [0,1]$ measures the intensity of the interest of user $u_m \in \mathcal{U}$ for concept $c_k \in \mathcal{O}$ (a class or an instance) in the domain ontology \mathcal{O} , K being the total number of concepts in the ontology. Similarly, the items $d_n \in \mathcal{D}$ in the retrieval space are assumed to be described (annotated) by vectors $\mathbf{d}_n = (d_{n,1}, d_{n,2}, \dots, d_{n,K})$ in the same vector-space as user preferences. Based on this common logical representation, measures of user interests for content items can be computed by comparing preference and annotation vectors, and they can be used to prioritise, filter and rank contents (a catalogue, a search result) in a personal way. Figure 1 shows our twofold-space ontology-based knowledge representation.

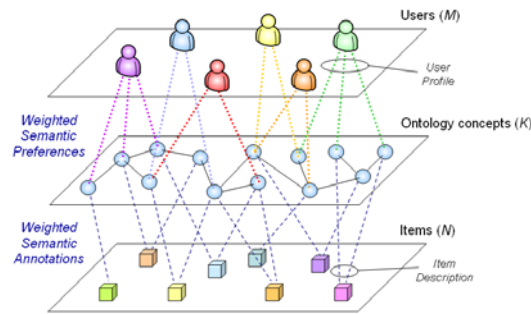


Figure 1. Ontology-based item description and user profile representations

The ontology-based representation is richer and less ambiguous than a keyword-based or item-based model. It provides an adequate grounding for the representation of coarse to fine-grained user interests (e.g. interests for a sports team, an actor). An ontology provides further formal, computer-processable meaning on the concepts (who is coaching a team, an actor’s filmography), and makes it available for a personalisation system to take advantage of. Further, ontology standards, such as OWL¹, support

¹ <http://www.w3.org/TR/owl-ref/>

inference mechanisms that can be used to enhance personalisation, so that, for instance, a user interested in *animals* (superclass of *cat*) is also recommended items about *cats*. Inversely, a user interested in *lizards*, *chameleons* and *snakes* can be inferred with a certain confidence to be interested in *reptiles*. Also, a user keen on *Spain* can be assumed to like *Madrid*, through the *locatedIn* relation, assuming that this relation has been seen as relevant for inferring previous user interests.

4. Ontology-based Personalised Content Retrieval

Our retrieval model (wrapped by the ‘Item Retrieval’ component in Figure 2) works in two phases. In the first one, a formal ontology-based query (e.g. in RDQL¹) is issued by some form of query interface (e.g. NLP-based) which formalises an information need. The query is processed against the knowledge base using any desired inference or query execution tool, outputting a set of ontology concepts that satisfy the query. From this point, the retrieval model is based on an adaptation of the classic vector IR model, where the axes of the vector space are the concepts of \mathcal{O} , instead of text keywords. Like in the classic model, in ours the query and each item are represented by vectors \mathbf{q} and \mathbf{d} , so that the degree of satisfaction of a query by an item can be computed by the cosine measure $sim(d, q) = \cos(\mathbf{d}, \mathbf{q})$. The problem, of course, is how to build the \mathbf{d} and \mathbf{q} vectors. For more details, see (Castells, Fernández & Vallet, 2007). Here, we obviate this issue, and continue explaining our content retrieval process with its personalisation phase (component ‘Personalised Ranking’ in Figure 2). Our notion of personalisation is based on the definition of a relevance measure $pref(d, u) = \cos(\mathbf{d}, \mathbf{u})$ of an item d for a user u . In order to bias the result of a search to the preferences of the user, the above measure has to be combined with the query-based score without personalisation $sim(d, q)$ defined previously, to produce a combined ranking.

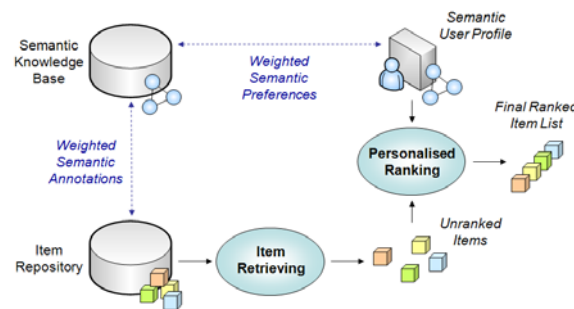


Figure 2. Architecture of the personalised semantically annotated item retrieval process

¹ <http://www.w3.org/Submission/RDQL/>

In real scenarios, profiles tend to be scattered. Users are usually not willing to spend time describing their detailed preferences to the system, and applications where automatic preference learning is applied tend to recognise main characteristics of user preferences, thus yielding profiles that may entail a lack of expressivity. To overcome this problem, we propose a preference spreading mechanism, which expands the initial preferences stored in user profiles through explicit semantic relations of the ontology (Figure 3). Based on Constrained Spreading Activation, CSA, (Cohen & Kjeldsen, 1987), the expansion is self-controlled by applying a decay factor to the intensity of preference each time a relation is traversed, and constraints (threshold weights) as stop criterion. In the conducted experiments, we shall show that this spreading provides a better performance on individual personalisation, and is essential for our CoI identification method.

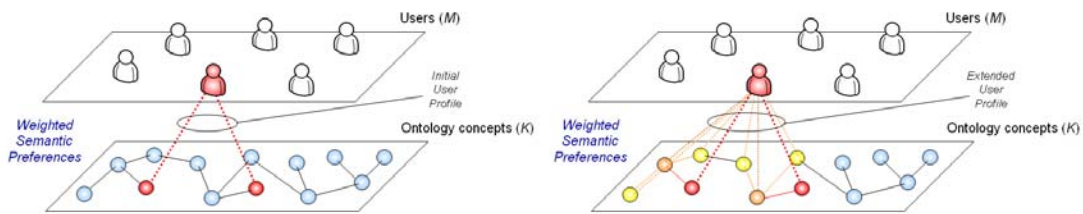


Figure 3. Preference expansion of a semantic user profile

5. Multilayered Semantic Communities of Interest

In social communities, it is commonly accepted that people who are known to share a specific interest are likely to have additional connected interests. For instance, people who share interests in travelling might be also keen on topics related in photography, gastronomy or languages. In fact, this assumption is the basis of most recommender systems (Adomavicius & Tuzhilin, 2005). We assume this hypothesis here as well.

We propose to exploit the links between users and concepts to extract relations among users and derive semantic social networks according to common interests. Analysing the structure of the domain ontology and taking into account the preference weights of the user profiles we shall cluster the domain concept space generating groups of interests shared by several users. Thus, those users who share interests of a specific concept cluster will be connected in the network, and their preference weights will measure their degree of membership to each cluster. Specifically, a vector $\mathbf{c}_k = (c_{k,1}, c_{k,2}, \dots, c_{k,M})$ is assigned to each concept $c_k \in \mathcal{O}$, where $c_{k,m} = u_{m,k}$ is the weight of concept c_k in the semantic profile of user u_m . Based on these vectors a classic hierarchical clustering strategy is

applied. The obtained clusters (Figure 4) represent the groups of preferences (topics of interests) in the concept-user vector space shared by a significant number of users.

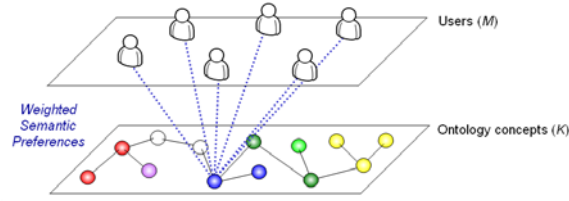


Figure 4. Semantic concept clustering based on the shared interests of the users

Once the concept clusters are created, each user can be assigned to a specific cluster. The similarity between a user's preferences $\mathbf{u}_m = (u_{m,1}, u_{m,2}, \dots, u_{m,K})$ and a cluster C_q is computed by $sim(\mathbf{u}_m, C_q) = |C_q|^{-1} \sum_{c_k \in C_q} u_{m,k}$, where c_k represents the concept that corresponds to the $u_{m,k}$ component of the user preference vector, and $|C_q|$ is the number of concepts included in the cluster. The clusters with highest similarities can then be assigned to the users, thus creating groups of users with shared interests (Figure 5).

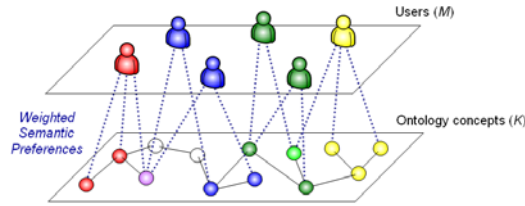


Figure 5. Groups of users obtained from the semantic concept clusters

The concept and user clusters can be used to find emergent, focused semantic Communities of Interest (CoI). The preference weights of the user profiles, the degrees of membership of the users to each cluster, and the similarity measures between clusters are used to find relations between two distinct types of social items: individuals and groups of individuals. Taking into account the concept clusters, user profiles are partitioned into semantic segments. Each of these segments corresponds to a concept cluster, and represents a subset of the user interests that is shared by the users. By thus introducing further structure in profiles, it is now possible to define relations among users at different levels, obtaining a multilayered interest network. Figure 6 illustrates this idea. The image on the left represents a situation where four clusters are obtained. Based on them (images on the right), user profiles are partitioned in four semantic layers. On each layer, weighted relations among users are derived, building up different semantic CoI.

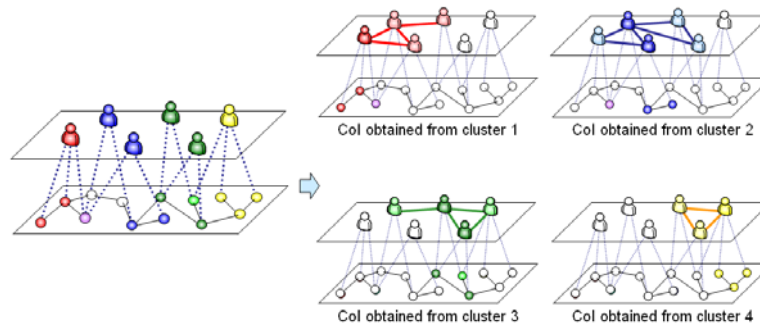


Figure 6. Multilayered semantic CoI built from the obtained clusters

The resulting CoI have many potential applications. For example, they can be exploited to the benefit of content-based collaborative filtering recommendations, not only because they establish similarities between users, but also because they provide powerful means to focus on different semantic contexts for different information needs. The design of information retrieval models in this direction is explored in Section 7. Additionally, the identified user clusters can be utilised for group profile modelling. In the next section, we propose several user profile merging strategies to build group profiles reflecting human voting criteria when a choosing of an item has to be made.

6. Ontology-based Group Profiles for Content Retrieval

While the creation and exploitation of individual models of user preferences and interests have been largely explored by the user modelling and personalisation research community, group modelling - combining individual user models to model a group - has not received the same attention (Ardissono *et al.*, 2003).

The question that arises is how a system can adapt itself to a collective, in such a way that each individual enjoys or even benefits from the results. Though explicit group preference modelling has been addressed to a rather limited extent in prior work in the computing field, the related issue of social choice, i.e. deciding what is best for a group given the opinions of individuals, has been studied extensively in economics, politics, sociology, and mathematics (Taylor, 1995). The models of a social welfare function in these works are similar to the group modelling problem we put forward here.

In this work, we study the feasibility of applying strategies, based on social choice theory (Masthoff, 2004), for combining multiple individual preferences in the personalisation framework explained in Section 4, and using the semantic CoI obtained with the user clustering strategy described in Section 5. Combining a set of ontology-based user profiles, our framework retrieves ranked lists of items according to the interests and preferences of the members of the group (see Figure 7).

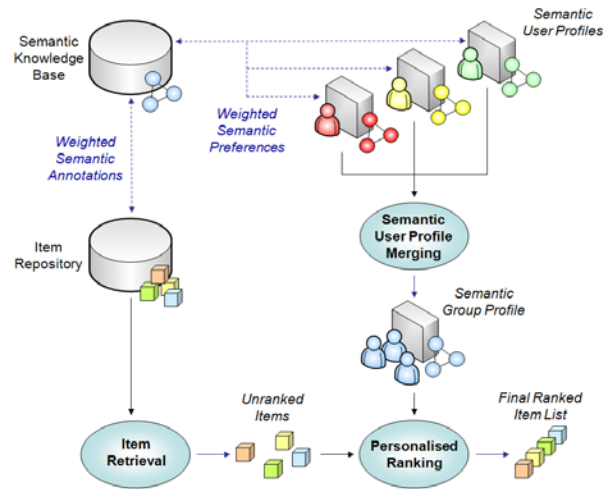


Figure 7. Architecture of the group profile-based semantic annotated item retrieval process

6.1. Group Modelling Strategies

In (Masthoff, 2004), the author discusses several techniques for combining individual user models to adapt to groups. Considering a list of TV programs and a group of viewers, she investigates how humans select a sequence of items for the group to watch. In our context, we slightly modified those techniques. For instance, due to preference weights have to belong to the range $[0,1]$, the weights obtained for a given group profile are normalised. The following are brief descriptions of the strategies.

- **Additive Utilitarian Strategy.** Preference weights from all the users of the group are added, and the larger the sum the more influential the preference is for the group.
- **Multiplicative Utilitarian Strategy.** Instead of adding the preference weights, they are multiplied, and the larger the product the more influential the preference is for the group. This strategy could be self-defeating: in a small group the opinion of each individual will have too much large impact on the product.
- **Borda Count.** Scores are assigned to the preferences according to their weights in a user profile: those with the lowest weight get zero scores, the next ones up one point, and so on. When multiple preferences have the same weight, the averaged sum of their hypothetical scores is equally distributed to the involved preferences.
- **Copeland Rule.** This strategy sorts the preferences according to their *Copeland* index: the difference between the number of times a preference beats (has higher weights) the rest of the preferences and the number of times it loses.
- **Approval Voting.** Only those weights greater or equal than a given threshold (e.g. 0.5) are taking into account for the profile combination. A preference receives a

vote for each user profile that has its weight surpassing the establish threshold. The larger the number of votes the more influential the preference is for the group.

- **Least Misery Strategy.** The weight of a preference in the group profile is the minimum of its weights in the user profiles. The lower weight the less influential the preference is for the group. A group is as satisfied as its least satisfied member.
- **Most Pleasure Strategy.** It works as the Least Misery Strategy, but instead of considering for a preference the smallest weights of the users, it selects the greatest ones. The higher weight the more influential the preference is for the group.
- **Average without Misery Strategy.** As the Additive Utilitarian Strategy, this one assigns a preference the average of the weights in the individual profiles. The difference here is that those preferences which have a weight under a certain threshold (we used 0.25) will not be considered.
- **Fairness Strategy.** The top preferences from all the users of the group are considered. We have decided to select only the $L/2$ best ones, where L is the number of preferences not assigned to the group profile yet. From them, the preference that least misery causes to the group (that from the worst alternatives that has the highest weight) is chosen for the group profile with a weight equal to L . The process continues in the same way considering the remaining $L-1$, $L-2$, etc. preferences and uniformly diminishing to 0 the further assigned weights.
- **Plurality Voting.** This method follows the same idea of the Fairness Strategy, but instead of selecting from the $L/2$ top preferences the one that least misery causes to the group, it chooses the alternative which most votes have obtained.

6.2. Experiments

The experiments done try to find the group modelling strategy that best fits the human way of selecting items when personal tastes of a group have to be taken into consideration. The scenario was the following. A set of twenty four pictures was considered. For each picture several annotations were set, describing their topics (at least one of *beach*, *construction*, *family*, *vegetation*, and *motor*) and the degrees (real numbers in $[0,1]$) of appearance these topics have on the picture. Twenty subjects participated in the experiments. They were Ph.D. students of our department. They were asked to think about a group of three users with different tastes. In decreasing order of preference (i.e., progressively smaller weights): a) User₁ liked *beach*, *vegetation*, *motor*,

construction and *family*, b) User₂ liked *construction*, *family*, *motor*, *vegetation* and *beach*, and c) User₃ liked *motor*, *construction*, *vegetation*, *family* and *beach*.

We have defined a distance that measures the existing difference between two given ranked item lists. The goal is to determine which group modelling strategies give ranked lists closest to those empirically obtained from several subjects. Consider \mathcal{D} as the set of items stored and retrieved by the system. Let $\tau_{sub} \in [0,1]^N$ be the item ranked list for a given subject and let $\tau_{str} \in [0,1]^N$ be the item ranked list for a specific combination strategy, where N is the number of items stored by the system. We use the notation $\tau(d)$ to refer the position of the item $d \in \mathcal{D}$ in the ranked list τ .

The distance $d(\tau_{sub}, \tau_{str}) = \sum_{d \in \mathcal{D}} |\tau_{sub}(d) - \tau_{str}(d)|$ might represent a good measure of the disparity between a subject's preference list and the ranked list obtained from a group modelling strategy. This expression basically sums the differences between the positions of each item in the subject and strategy ranked lists. The smaller the distance the more similar the ranked lists. However, in typical information retrieval systems, a user usually takes into account only the first top ranked items. In general, he will not browse the entire list of results, but stop at some top n in the ranking. Thus, we propose to more consider those items that appear before the n -th position of the strategy ranking and after the n -th position of the subject ranking, in order to penalise more those of the top n items in the strategy ranked list that are not relevant for the user:

$$d(\tau_{sub}, \tau_{str}) = \sum_{n=1}^N P(n) \frac{1}{n} \sum_{d \in \mathcal{D}} |\tau_{sub}(d) - \tau_{str}(d)| \chi_n(d, \tau_{sub}, \tau_{str})$$

where $P(n)$ is the probability that the user stops browsing the list at position n , and

$$\chi_n(d, \tau_{sub}, \tau_{str}) = \begin{cases} 1 & \text{if } \tau_{str}(d) \leq n \text{ and } \tau_{sub}(d) > n \\ 0 & \text{otherwise} \end{cases}.$$

The problem here is how to define the probability $P(n)$. Although an approximation to the distribution function for $P(n)$ can be taken e.g. by interpolation of data from a statistical study, we simplify the model fixing $P(10)=1$ and $P(n)=0$ for $n \neq 10$, assuming that users are only interested in those items shown at first time after a query:

$$d_{10}(\tau_{sub}, \tau_{str}) = \frac{1}{10} \sum_{d \in \mathcal{D}} |\tau_{sub}(d) - \tau_{str}(d)| \chi_{10}(d, \tau_{sub}, \tau_{str}) \quad (1)$$

Observing the twenty four pictures, and taking into account the preferences of the three users belonging to the group, the subjects were asked to make an ordered list of the pictures. With these lists we measure the distance d_{10} with respect to the ranked lists given by the group modelling strategies. The average results are shown in Figure 8. From the figure, it seems that strategies like *Borda Count* and *Copeland Rule*, which do not depend on specific thresholds, give lists more similar to those manually created by the subjects, and strategies such as *Average without Misery* and *Plurality Voting* obtain the greatest distances. Additionally, we compare the strategy lists against those obtained using user profiles. Surprisingly, their distances are very similar to the empirical ones. They agree with the strategies that seem to be more adequate for group modelling.

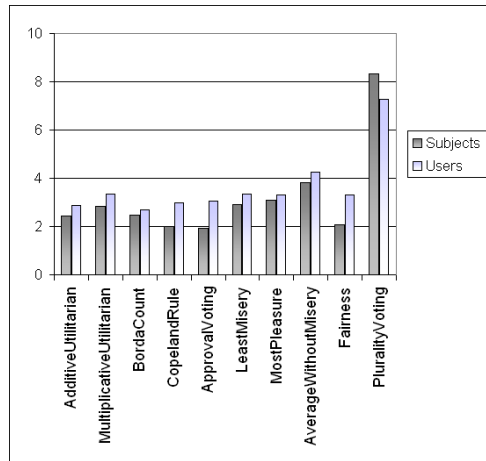


Figure 8. Average distance d_{10} between the ranked lists obtained with the combination strategies, and the lists created by the subjects and the lists retrieved using the individual semantic user profiles

7. Semantic Content-based Collaborative Recommendations

Collaborative filtering adapt to groups of people who interact with the system, so that single users take advantage from the experience of other users with which they have certain traits or interests in common. User groups may be quite heterogeneous, and it might be very difficult to define the methods for which the system adapts itself to the groups. We believe that exploiting the relations of the underlying CoI which emerge from the users' interests can offer an important benefit. Using our CoI proposal (Section 5), we present recommenders that generate ranked item lists in different contexts.

7.1. Recommendation Models

Our collaborative framework is based on the profile of the user to whom the ranked list is delivered, and represents the situation where his interests are compared to those of other users in a social network. We consider two different approaches: a) one that

generates a unique ranked list based on the similarities between the items and all the existing semantic clusters, and, b) one that provides a ranking for each semantic cluster. Thus, we study two recommendation strategies, UP (profile-based), and UP- q (profile-based, considering a specific cluster C_q), which are formalised next. For a user profile \mathbf{u}_m , an information item vector \mathbf{d}_n , and a cluster C_q , we denote by \mathbf{u}_m^q and \mathbf{d}_n^q the projections of the corresponding concept vectors onto cluster C_q , i.e. the k -th component of \mathbf{u}_m^q and \mathbf{d}_n^q is $u_{m,k}$ and $d_{n,k}$ respectively if $c_k \in C_q$, and 0 otherwise.

- **Model UP.** The semantic profile of a user u_m is used by the system to return a unique ranked list. The preference score of an item d_n is computed as a weighted sum of the indirect preference values based on similarities with other users in each cluster. The sum is weighted by the similarities with the clusters, as follows:

$$pref(d_n, u_m) = \sum_q nsim(d_n, C_q) \sum_i nsim_q(u_m, u_i) \cdot sim_q(d_n, u_i) \quad (2)$$

where:

$$sim(d_n, C_q) = \frac{\sum_{c_k \in C_q} d_{n,k}}{\|\mathbf{d}_n\| \sqrt{|C_q|}}, \quad nsim(d_n, C_q) = \frac{sim(d_n, C_q)}{\sum_i sim(d_n, C_i)}$$

are the single and normalised similarities between the item d_n and the cluster C_q ,

$$sim_q(u_m, u_i) = \cos(\mathbf{u}_m^q, \mathbf{u}_i^q) = \frac{\mathbf{u}_m^q \cdot \mathbf{u}_i^q}{\|\mathbf{u}_m^q\| \cdot \|\mathbf{u}_i^q\|}, \quad nsim_q(u_m, u_i) = \frac{sim_q(u_m, u_i)}{\sum_j sim_q(u_m, u_j)}$$

are the single and normalised similarities at layer q between users u_m and u_i , and

$$sim_q(d_n, u_i) = \cos(\mathbf{d}_n^q, \mathbf{u}_i^q) = \frac{\mathbf{d}_n^q \cdot \mathbf{u}_i^q}{\|\mathbf{d}_n^q\| \cdot \|\mathbf{u}_i^q\|}$$

is the similarity at layer q between item d_n and user u_i .

- **Model UP- q .** The user's preferences are used by the system to return one ranked list per cluster, obtained from the similarities between users and items at each cluster layer. The ranking that corresponds to the cluster for which the user has the highest membership value is selected. The expression is analogous to equation (2), but it does not include the term that connects the item with each cluster C_q .

$$pref_q(d_n, u_m) = \sum_i nsim_q(u_m, u_i) \cdot sim_q(d_n, u_i) \quad (3)$$

7.2. Experiments

We have performed an experiment with real subjects in order to evaluate the effectiveness of our recommendation models. The experiment was setup as follows. The set of twenty four pictures used in Section 6 was again considered as the retrieval space. Each picture was annotated with semantic metadata describing what the image depicts, using an ontology with concepts of six topics: *animals*, *beach*, *construction*, *family*, *motor* and *vegetation*. A weight in $[0,1]$ was assigned to each annotation, reflecting the relative importance of a concept in the picture. Twenty Ph.D. students of our department were asked to independently define their weighted preferences about a list of concepts related to the above topics. No restriction was imposed on the number of topics and concepts to be selected by each of the students. Indeed, the generated profiles had very different characteristics, observable not only in their joint interests, but also in their complexity.

Once the profiles were created, we run our method. After the execution of the preference spreading, the concept space was clustered according to similar user interests. In this phase, because our strategy is based on a hierarchical clustering method, various clustering levels were found, expressing different compromises between complexity, described in terms of number of clusters, and compactness, defined by the number of concepts per cluster. Table 6 shows the concept clusters obtained at clustering level $R = 4$. We underline those general concepts that initially did not appear in the profiles. Inferred from our preference spreading strategy, these concepts helped to build the CoI.

Table 1. Concept clusters obtained at clustering level $R=4$

Cluster	Concepts
1	ANIMALS: Rabbit CONSTRUCTION: <u>Construction</u> , Speedway, Racing-Circuit, Short-Oval, Garage, Lighthouse, Pier, Beach-Hut, Mountain-Shelter, Mountain-Villa, Mountain-Hut, MOTOR: <u>Vehicle</u> , Ambulance, Racing-Car, Tractor, Canoe, Surf, Windsurf, Water-Motor, Sleigh, Snow-Cat, Lift, Chair-Lift, Toboggan, Cable-Car
2	ANIMALS: <u>Organism</u> , <u>Agentive-Physical-Object</u> , Reptile, Snake, Tortoise, Sheep, Dove, Fish, Mountain-Goat, Reindeer CONSTRUCTION: <u>Non-Agentive-Physical-Object</u> , <u>Geological-Object</u> , <u>Ground</u> , <u>Artefact</u> , Fortress, Road, Street FAMILY: <u>Civil-Status</u> , Wife, Husband MOTOR: <u>Conveyance</u> , Bicycle, Motorcycle, Helicopter, Boat, Sailing-Boat
3	ANIMALS: <u>Animal</u> , <u>Vertebrates</u> , <u>Invertebrates</u> , <u>Terrestrial</u> , <u>Mammals</u> , Dog, ‘Tobby’ (instance of Dog), Cat, Horse, Bird, Eagle, Parrot, Pigeon, Butterfly, Crab BEACH: Water, Sand, Sky VEGETATION: <u>Vegetation</u> , ‘Tree’ (instance of Vegetation), ‘Plant’ (instance of Vegetation), ‘Flower’ (instance of Vegetation)
4	FAMILY: <u>Family</u> , Grandmother, Grandfather, Parent, Mother, Father, Sister, Brother, Daughter, Son, Mother-In-Law, Father-In-Law, Cousin, Nephew, Widow, ‘Fred’ (instance of Parent), ‘Christina’ (instance of Sister), ‘Peter’ (instance of Brother)

Cluster 1 contains most of the concepts related to *construction* and *motor*, showing a significant correlation between these topics of interest. Cluster 2 is the one with more different topics and general concepts. It is also notorious that the concepts ‘wife’ and ‘husband’ appear in the cluster. This is due to these concepts were not be annotated by the subjects, who were students, not married at the moment. Cluster 3 is the one that gathers all the concepts about *beach* and *vegetation*. The subjects who liked vegetation also seemed to be interested in beaches. It also has many of the concepts belonging to the topic of *animals*, but in contrast to cluster 2, the annotations were for more common and domestic animals. Finally, cluster 4 collects most of the *family* concepts.

We evaluated the retrieval models computing their average precision/recall curves assigning to the existing clusters their most similar users. Figure 9 exposes the results at clustering level $R = 4$. The version UP- q seems to outperform the version UP. Obviously, the more clusters we have, the better performance is achieved, as they tend to have assigned fewer users and be more similar to the individual profiles. However, it can be seen that good results are obtained with only three clusters. Additionally, for both models, we plot with dotted lines the curves achieved without spreading user preferences. It can be pointed out that our strategy performs better when it is combined with CSA.

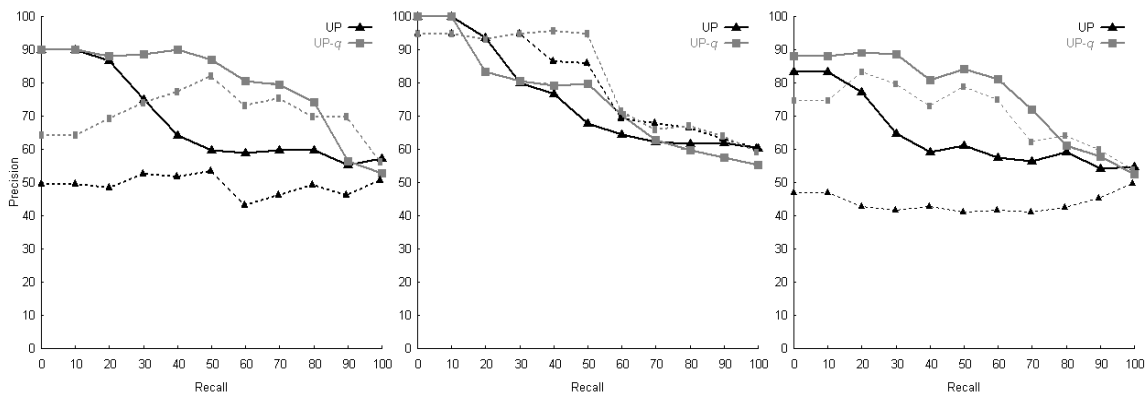


Figure 9. Average precision vs. recall curves for users assigned to the user clusters obtained with the UP (black lines) and UP- q (grey lines) models at level $R=4$ concept clusters

8. Discussion

In this work, we have presented an approach to the automatic identification of CoI according to ontology-based user profiles. Taking into account the users’ preferences we cluster the ontology concept space, obtaining common topics of interest. With these topics, preferences are partitioned into different layers. The degree of membership of the obtained sub-profiles to the clusters, and the similarities among them are used to define links exploited by group modelling and collaborative filtering techniques.

Our implementation of the applied clustering strategy was a hierarchical procedure. We have to study more efficient clustering algorithms, e.g. based on Latent Semantic Analysis (Deerwester *et al.*, 1990). We are also aware of the need to test our approach in combination with automatic preference learning techniques in order to investigate its robustness to imprecise user interests, and the impact of the accuracy of the ontology-based profiles on the correct performance of the clustering processes.

References

- Adomavicius, G., & Tuzhilin, A. (2005). Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Transactions on Knowledge and Data Engineering* 17(6), pp. 734-749.
- Alani, H., O'Hara, K., & Shadbolt, N. R. (2002). ONTOCOPI: Methods and Tools for Identifying Communities of Practice. *In Proceedings of the 17th IFIP World Computer Congress*, pp. 225-236.
- Ardissono, L., *et al.* (2003). INTRIGUE: Personalized Recommendation of Tourist Attractions for Desktop and Handset Devices. *Applied AI* 17(8-9), pp. 687-714.
- Castells, P., Fernández, M., & Vallet, D. (2007). An Adaptation of the Vector-Space Model for Ontology-based Information Retrieval. *IEEE Transactions on Knowledge and Data Engineering* 19(2), pp. 261-272.
- Cohen, P. R., & Kjeldsen, R. (1987). Information Retrieval by Constrained Spreading Activation in Semantic Networks. *Inf. Processing and Management* 23(4), pp. 255-268.
- Deerwester, S., *et al.* (1990). Indexing by Latent Semantic Analysis. *Journal of the Society for Information Science* 41(6), pp. 391-407.
- Liu, H., Maes, P., & Davenport, G. (2006). Unravelling the Taste Fabric of Social Networks. *Int. Journal on Semantic Web and Information Systems* 2(1), pp. 42-71.
- Masthoff, J. (2004). Group Modelling: Selecting a Sequence of Television Items to Suit a Group of Viewers. *User Modelling and User-Adapted Interaction* 14(1), pp. 37-85.
- Middleton, S. E., Roure, D. D., & Shadbolt, N. R. (2004). Ontology-based Recommender Systems. *In Handbook on Ontologies*, pp. 477-498. Springer.
- Mika, P. (2005). Flink: Semantic Web Technology for the Extraction and Analysis of Social Networks. *Journal of Web Semantics* 3(2-3), pp. 211-223.
- Taylor, A. (1995). *Mathematics and Politics: Strategy, Voting, Power and Proof*. Springer.
- Wenger, E. (1998). *Communities of Practice: Learning, Meaning and Identity*. Cambridge University Press.