
A Multi-Purpose Ontology-Based Approach for Personalised Content Filtering and Retrieval

Iván Cantador¹, Miriam Fernández¹, David Vallet¹, Pablo Castells¹, Jérôme Picault², and Myriam Ribière²

¹ Escuela Politécnica Superior, Universidad Autónoma de Madrid, 28048 Madrid, Spain, ivan.cantador@uam.es, miriam.fernandez@uam.es, david.vallet@uam.es, pablo.castells@uam.es

² Motorola Labs, Parc Les Algorithmes, St. Aubin, 91193 Gif-sur-Yvette, France, jerome.picault@motorola.com, myriam.ribiere@motorola.com

Summary. Personalised multimedia access aims at enhancing the retrieval process by complementing explicit user requests with implicit user preferences. We propose and discuss the benefits of the introduction of ontologies for an enhanced representation of the relevant knowledge about the user, the context, and the domain of discourse, as a means to enable improvements in the retrieval process and the performance of adaptive capabilities. We develop our proposal by describing techniques in several areas that exemplify the exploitation of the richness and power of formal and explicit semantics descriptions, and the improvements therein. In addition, we discuss how those explicit semantics can be learnt automatically from the analysis of the content consumed by a user, determining which concepts appear to be significant for the user's interest representation. The introduction of new preferences on the user profile should correspond to heuristics that provide a trade-off between consistency and persistence of the user's implicit interests.

1 Introduction

Personalised multimedia access aims at enhancing the retrieval process by complementing explicit user requests with implicit user preferences, to better meet individual user needs [9]. Automatic user modelling and personalisation has been a thriving area of research for nearly two decades, gaining significant presence in commercial applications around the mid-90s. Popular online services such as Google [1, 37] or Amazon [26, 32] are nowadays exploiting some personalisation features, in particular to improve their content retrieval systems. Even if these systems have the merit of having been deployed at a large scale, they rely on rather simple models, which may often be inaccurate or still provide results that do not completely match

users' expectations. Indeed, personalising a content retrieval system involves considerable complexity, mainly because finding implicit evidence of user needs and interests through their behaviour is not an easy task. This difficulty is often considerably increased by an imprecise and vague representation of the semantics involved in user actions and system responses, which makes it even more difficult to properly pair user interests and content descriptions. The ambiguity of terms used in this representation, the unclear relationships between them, their heterogeneity, especially in current ever-growing large-scale networked environments such as the WWW, often constitute a major obstacle for achieving an accurate personalisation, e.g. when comparing user preferences to content items, or users among themselves.

In this chapter we argue for the introduction of ontologies [17] as an enhanced representation of the relevant knowledge about the domain of discourse, about users, about contextual conditions, involved in the retrieval process, as a means to enable significant improvements in the performance of adaptive content retrieval services. We illustrate our point by describing the development of advanced features and enhancements in specific areas related to personalisation where the ontology-based approach shows its benefit, including:

- Basic personalised content search and browsing based on user preferences;
- Learning semantic user preferences over time;
- Dynamic contextualisation of user preferences;
- Dynamic augmented social networking and collaborative filtering.

Domain ontologies and rich knowledge bases play a key role in the models and techniques that we propose in the above areas, as will be described in the sequel. The approaches presented in this chapter share and exploit a common representation framework, thus obtaining multiple benefits from a shared single ontology-rooted grounding. Furthermore, it will be shown that modular semantic processing strategies, such as inference, graph processing, or clustering, over networked ontology concepts, may be reused and combined to serve multiple purposes.

The rest of the chapter is organized as follows. Section 2 introduces the basic approach for the ontology-oriented representation of semantic user preferences, and its application to personalised content search and retrieval. Following this, Sect. 3 explains how these semantic user preferences may evolve automatically over time. Then Sect. 4 describes an approach for the dynamic contextualisation of semantic user preferences, and Sect. 5 shows the extension of the techniques described in previous sections to multi-user environments, based on collaborative personalisation strategies. Finally, some conclusions are given in Sect. 6.

2 Ontology-Based Personalisation for Content Retrieval

A very significant body of research in the last two decades has paid attention to the problem of personalising information access and delivery, commonly addressed under such names as information filtering, collaborative filtering, or personalised information retrieval, with variations in approach and perspective [16, 18, 20]. Formulated and approached from different angles, the problem has been a major research topic in the information retrieval, user modelling, and machine learning fields. The research activity in this area has been paralleled by a comparable interest towards making such techniques commercially profitable [1, 26, 32, 37].

Most of the proposed techniques in this area keep and process long records of accessed documents by each user, in order to infer potential preferences for new documents (e.g. by finding similarities between documents, or between users). The data handled by these techniques have been rather low-level and simple: document IDs, text keywords and topic categories at most [21, 28]. In contrast, the recent proposals and achievements towards the enrichment of text and multimedia content by formal, ontology-based, semantic descriptions open new opportunities for improvement in the personalisation field from a new, richer representational level [4, 9]. We see indeed the introduction of ontology-based technology in the area of personalisation as a promising research direction [14]. Ontologies enable the formalisation of user preferences in a common underlying, interoperable representation, whereby user interests can be matched to content meaning at a higher level, suitable for conceptual reasoning.

An ontology-based representation is richer, more precise, and less ambiguous than a keyword-based model. It provides an adequate grounding for the representation of coarse user interests (e.g. for sports, cinema, finance, baseball, sci-fi movies, oil industry) to fine-grained preferences (e.g. for individual items such as a sports team, an actor, a stock value) in a hierarchical way, and can be a key enabler to deal with the subtleties of user preferences. An ontology provides further formal, computer-processable meaning on the concepts (e.g. who is coaching a team, an actor's filmography, financial data on a stock), and makes it available for the personalisation system to take advantage of. Moreover, an ontology-rooted vocabulary can be agreed and shared (or mapped) between different systems, or different modules of the same system, and therefore user preferences, represented this way, can be more easily shared by different players. For instance, a personalisation framework may share a domain ontology with a knowledge-based content analysis tool that extracts semantic metadata from audio/visual content, conforming to the ontology [4]. On this basis, it is easier to build algorithms that match preference to content, through the common domain ontology.

In an ontology-based approach, semantic user preferences may be represented as a vector of weights (numbers from -1 to 1), representing the intensity of the user interest for each concept, where negative values indicative

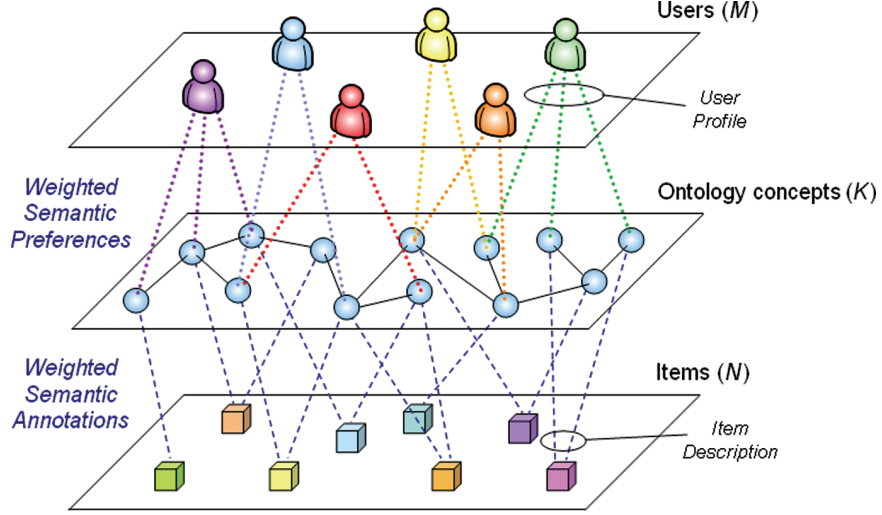


Fig. 1. Association of users and content

of a dislike for that concept [9]. Similarly, content can be described by a set of weighted concepts (values from 0 to 1, indicating the intensity of relation between the content and the concept) in such a way that users can be accurately related through the ontology layer to the content units in the search space (see Fig. 1).

If a content analysis tool identifies, for instance, a cat in a picture, and the user is known to like cats, the personalisation module can make predictions on the potential user interest for the picture by comparing the metadata of the picture, and the preferred concepts in the user profile. Furthermore, ontology standards backed by international consortiums (such as the W3C), and the corresponding available processing tools, support inference mechanisms that can be used to further enhance personalisation, through the middle ontology layer, so that, for instance, a user interested in animals (superclass of cat) is also recommended pictures of cats. Inversely, a user interested in lizards, snakes, and chameleons can be inferred to be interested in reptiles with a certain confidence. Also, a user keen of Sicily can be supposed to like Palermo, through the transitive *locatedIn* relation, assuming that this relation has been seen as relevant for inferring previous underlying user's interests. In fact, it is even possible to express complex preferences based on generic conditions, such as “athletes that have won a gold medal in the Olympic Games”.

Based on preference weights, measures of user interest for content units can be computed, with which it is possible to discriminate, prioritize, filter and rank contents (a collection, a catalogue section, a search result) in a personal way.

The basis for the personalisation of content retrieval is the definition of a matching algorithm that provides a *personal relevance measure* (PRM) of a

content object d for a particular user u , according to his/her semantic preferences. The measure is computed as a function of the semantic preferences of u and the semantic metadata of d . In this calculation, user preferences and content metadata are seen as two vectors in a K -dimensional vector space, where K is the number of elements in the universe \mathcal{O} of ontology terms, and the coordinates of the vectors are the weights assigned to ontology terms in user preferences and content annotations. The weights represent the intensity of preference by u for each concept, and the degree of importance of each concept in the meaning conveyed by d , respectively. The PRM is thus represented as the algebraic similarity between the user preferences and the content vector. Using the classic Information Retrieval vector-space model [2], this similarity can be measured by the cosine function. Note that in our approach the preference vector plays an equivalent role to the query vector in classic IR. Of course, the accuracy of the PRM comparison is directly dependent on the accuracy of content annotations and the represented user preferences.

Figure 2 represents the similarity between two different items d_1 and d_2 , and the semantic preferences of the user u .

The PRM algorithm thus matches two concept-weighted vectors and produces a value between -1 and 1 . Values near -1 indicate that the preferences of the user do not match the content metadata (i.e. the two vectors are dissimilar); values near 1 indicate that the user interests do match the content. In cases where annotations are attached to the items in a Boolean way (because the underlying model or system does not provide for a finer weighting), a weight of 1 is assigned by default to all annotations.

Figure 3 shows an example where $\mathcal{O} = \{\text{beach}, \text{construction}, \text{family}, \text{vegetation}, \text{motor}\}$ is the set of all domain ontology terms (classes and instances). According to her profile, the user is interested in the concepts “beach”, “motor”, and “vegetation”, with different intensity, and has a negative preference for “construction”. The preference vector for this user is thus $\vec{u} = \{1.0, -0.7, 0.0, 0.2, 0.5\}$.

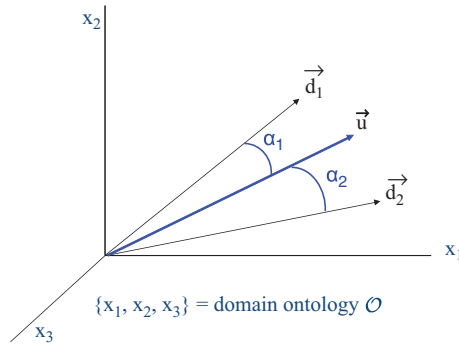


Fig. 2. Visual representation of metadata and preference's vector similarity

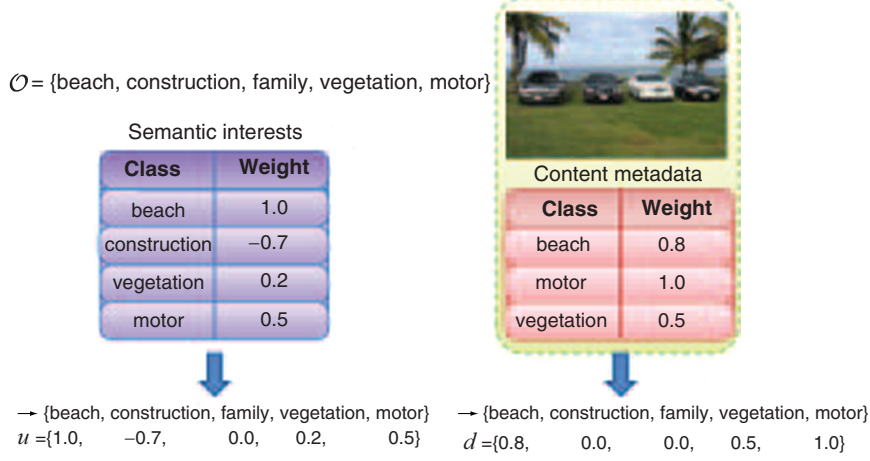


Fig. 3. Example of semantic preferences and metadata matching

A still image is annotated with the concepts “*beach*”, “*motor*” and “*vegetation*”, with the metadata vector $\vec{d} = \{0.8, 0.0, 0.0, 0.5, 1.0\}$.

The PRM of the still image for this user is thus $PRM = \cos(\vec{u}, \vec{d}) \approx 0.82^1$.

This measure can be combined with the relevance measures computed by user-neutral algorithms (e.g. a search result score), or other user-based algorithms like the ones introduced in the following sections, resulting in a personalised bias on the ranking of search results and/or content recommendations.

3 Adaptation of Semantic User Preferences

In the approach described in the previous section, personalised content retrieval relies on a description of the user preferences as a vector of concept weights. In the simplest approach, these weights can be set by the users themselves. However, this solution presents a number of drawbacks: in such a case, user preferences are often static, therefore do not mimic correctly the natural evolution of users’ tastes over time. In addition, setting manually preferences may take time, and requires some knowledge on the structure of the personalisation system, such as the exact meaning of concepts used to describe preferences, or the rating scale for weighting preferences, in order to minimize the introduction of self-induced bias [5]. An alternative to manually setting

¹ $\cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{|\vec{x}| \times |\vec{y}|} = \frac{\sum_{i=1}^n (x_i \times y_i)}{\sqrt{\sum_{i=1}^n (x_i^2)} \times \sqrt{\sum_{i=1}^n (y_i^2)}}$

preferences consist in analysing content retrieved and consumed by the user over a significant period of time to infer implicit associated preferences.

3.1 Automatic Creation and Update of Semantic Preferences

Thus, several methods exposed in various recent research [14, 15, 22] are exploiting collected data such as document representation, user feedback and some other metrics such as the time the user spent to read the document, the length of the document, the mean number of documents read by a user, etc. to create and update user preferences. Although most of these techniques rely on a keyword- or taxonomy-based description of user preferences, they can be applied and expanded for a richer ontology-based representation of user profiles.

Our approach consists in analysing the content consumed by a user to determine from the content annotations which concepts appear and how often they occur during a given time period. These content concepts (i.e. metadata) are compared to the user profile concepts (i.e. preferences). The process is slightly different depending if the content concepts appear or not in the user profile.

The introduction of new preferences on the user profile should correspond to heuristics that propose a trade-off between *consistency* (e.g. if the content consumed by the user deals with diverse, semantically-unrelated concepts or not) and *persistence* (i.e. how stable and recurrent content concepts are) of content metadata in the user's history. We can envision the following situations:

- A concept occurs once and its occurrence is confirmed with time with roughly the same level, this concept can be introduced after a period as a long-term preference of the user;
- A concept occurs once and its occurrence is very high on a short period, and then disappears very quickly. Even if this concept can be considered as a preference during a period of time, it must be removed very fast from the preferences, once the interest of the user is over;
- A concept occurs once but the occurrence is not very high and even if confirmed in time, it does not constitute a significant interest for the user. In that case this concept will never become a user preference;
- A concept occurs and becomes a preference as in the first case, but disappears with time. It must at a certain time be removed from the preferences.

To handle those use-cases, we propose to use a *concept history stack* as a mechanism to store all concepts representing potential user interests (because they occurred in the consumed contents).

The insertion of new preferences is an *off-line process* that uses the history stack to decide whether or not to add new preferences in the user profile. This process can be run periodically, for example at the end of the day, or at

any other frequency (which could be determined based on appropriate user studies).

The decision is based on the comparison between a value of *concept occurrence* $C_{occ} = N_{occ}/(D - d)^2$, computed during the off-line process, and a threshold P_{thd} for which a concept candidate of the stack history can become a new preference. Candidate concepts for which $C_{occ} > P_{thd}$, are introduced as new preferences. Their weight into the user profile is initialised to a default, neutral value (for example 0.5).

The removal of concepts from the profile can rely on two mechanisms. First, define a stack size limit: when a new concept is introduced as a possible preference candidate and the limit of the stack is reached, the concept with the lower C_{occ} is removed. Second, define a threshold R_{thd} so that when the value C_{occ} for a concept is under this threshold, the concept must be removed³.

The computation of C_{occ} is naturally reflecting a *decay factor* – or *gradual forgetting* of preferences – since the occurrence of a concept is divided by the age of the concept in the history stack. This decay factor is sufficient to handle gradual interest changes [24]. A main issue concerns user *shift of interest* [34], since an important interest of one day can potentially create a new preference in the profile that will take a week to disappear from the profile (based on the natural decay factor). Some solutions have been proposed to adapt the size of the time window which is considered for adaptation [25]. We propose to have an additional shorter time window to handle shift of interest, where the occurrence of the concept is followed day by day: if a user is showing a particular interest for a concept during a one day period, this concept will be quickly taken into account in preferences, but if this interest is not confirm the following day, we would like this preference to disappear quickly in the user profile. When this phenomenon is encountered in the process, an additional decay factor is added to the concept.

Those decay factors, are not impacting directly the weight of the user preferences, but are impacting the decision for keeping or not a preference in the profile, so that the weight of a concept still reflect the real user consumption of contents.

However, the adaptation of semantic user preferences does not only consists in adding or removing preferences, but also in updating the concept weights in the user profile, based on the analysis of consumed content. A possible mathematical model to apply for the update of concept weights, influenced by the proposal made by [30] is the following:

² Where N_{occ} is the number of time the concept occurs in the set of metadata of a consumed content (considered as relevant for the user); D is the date of the day when the process is launch; and d is the date of first appearance of the concept into the set of metadata of a consumed content.

³ Values of P_{thd} and R_{thd} have to be determined through experiments with real data.

$$w_{new} = w_{old} + fd * ContentRating * e^{-\beta * x * y} * \log \frac{time}{\log length}.$$

The w_{old} factor represents the current weight of the concept. fd is the relevance feedback factor given through an analysis of the content consumption; the relevance feedback can take either Boolean value or be multi-valued [6]. $ContentRating$ is the rank assigned to the content by the personalised retrieval system; it can use a cosine similarity measure between the content and the user profile, or any other measure implemented to rank content. The $\log \frac{time}{\log length}$ expression incorporates the time spent reading or watching a content item and the length of the content, which operates as the normalizing factor. The $e^{-\beta * x * y}$ factor is used to attend the personalised non-linear change of the concept's weight according to user usage history data. x represents the mean number of content that the user is consuming per day; y represents the number of consumed content where the concept appears in the set of metadata. The more content a user consumes per day, for example, the more slowly the weights increase. The β factor is a constant, which takes different values in the two opposite scenarios of consumed/non-consumed content. More precisely, in the case of non-consumed content, the changing rate (i.e. the decreasing rate) should be slower, since a non-consumed content does not constitute an explicit indication for non-interest. On the contrary, in case of consumed content the changing rate (i.e. the increasing rate) should be faster, since a read news item demonstrates a better indication for interest.

3.2 Exploitation of Semantic Links Between Concepts for Preferences Learning

We believe that the method described above can be significantly improved by taking into account the benefits of a domain ontology. We propose to exploit semantic links between concepts to provide two independent mechanisms:

- Interest assumption completion, which adds more potential user interests in the concept history stack, by using hierarchical and semantic relationships between concepts;
- Preference update expansion, which expands the re-weighting of a concept to the subset of correlated concepts.

Interest Assumption Completion

We propose to use semantic relationships between concepts as a mean to enhance the management of the concept history stack. The concept history stack expresses the set of assumptions on user interests. In their simplest form, those assumptions are only relying on the exact occurrence of the keywords or the concepts appearing in the content annotations. We propose to complement those conjectures by deducing additional interests from the semantic relationships expressed in the domain ontologies. Two types of semantic relationships

can be considered: (1) inheritance relationships and (2) semantic well-defined relationships (also called concept properties).

In the case of inheritance, the following example can be exploited: a user is interested in “cats”, we should deduce just by expansion of the concept “cat”, that the user is interested also to some extent in “animals”. By doing such assumption, the user will receive information that could not be directly related to cats. This assumption can be true, but it has to be confirmed by the occurrence of other subtypes of “animals”. Thus, if a new concept c_{new} appears in a user consumed content, its super-type $c_{supertype}$ is also introduced as a potential interest in the concept history stack, with a pseudo-occurrence⁴ value proportional to the occurrence of c : $N_{occ}(c_{supertype}) = \gamma_1 \times N_{occ}(c_{subtype})$, where $\gamma_1 < 1$, has to be determined empirically. This mechanism allows expanding the user preferences based on semantic knowledge of the concept, but avoids making any assumption *directly from the ontology* on user interests. Indeed, by doing this expansion of concepts into the concept history stack, the super-type can only be added into the user preferences when it has been confirmed enough by other sub-concepts so that its pseudo-occurrence reaches the insertion threshold P_{thd} . Following our example, if concepts “dog” and “horse” appear later in other consumed content items, they will confirm the assumption that the user is not only interested in cats, since the occurrence value of the super-type is increased each time one of its subtype appears in the concept history stack.

As ontologies are richer than simple taxonomies, the exploitation of other kinds of semantic relationships than inheritance can bring a significant value. Thus, if a concept c appears in a user consumed content, then we propose to introduce in the concept history stack all related concepts $c_{related}$ such as $\exists p_{j=1..m} \in \mathcal{P}, \exists c_{related} \in \mathcal{O} / p_j(c, c_{related})$, where \mathcal{O} is the set of ontology concepts and \mathcal{P} is the set of ontology properties. In that way, all concepts related directly (through semantic relationships) to a user interest (a concept) can be considered as candidates for becoming user preferences. This is done by setting a value for the pseudo-occurrence of related concepts: $N_{occ}(c_{related}) = \gamma_2 \times N_{occ}(c)$, where $\gamma_2 < 1$ is determined empirically.

By using knowledge represented in the ontology, we can enhance the preliminary acquisition of preferences, by selecting potential interests that are related to concepts appearing in the consumed content. By introducing such additional assumptions in the history stack, with factors to calculate the pseudo-occurrence of deduced concepts, we are just helping the system to acquire more quickly preferences, but we are not making direct assumptions on user preferences. This mechanism avoids risks of errors by always relying on the concept occurrence to confirm a possible interest.

⁴ Whereas the term “occurrence” refers to concepts that explicitly appear in the content, the term “pseudo-occurrence” applies to concepts that do not explicitly appear in the content, i.e. to concepts that “could have appeared” because of their semantic proximity with concepts present in the document.

Preference Update Expansion

In Sect. 3.1 concepts are learnt in isolation; the updated weight of an existing preference is only based on its own previous value, regardless of other influential related concepts. But, if concepts like “Mars”, “mission”, “NASA”, “spacecraft” appear, they should be counted as more significant than if seen in an isolated way, because there is a semantic recurrence, which may be detected by finding semantic paths between the concepts. The exploitation of the ontology knowledge may enable to take also into consideration in the preference weight update of a given concept the weight of semantically related concepts. For example, we want to update the weight of the concept c in the user preferences, and we know that this concept is semantically linked to at least another concept: $\exists c_{related} \in \mathcal{O}, \exists p_{j=1..n} \in \mathcal{P}/p(c, c_{related})$, then the new weight of c is updated as described in Sect. 3.1, and the new weight for each concept $c_{related}$ related to c can be computed thanks to the following formula: $w_{new}(c_{related}) = w_{old}(c_{related}) + sf_{c,c_{related}} \times w_{new}(c)$, where $w_{new}(c_{related})$ is the new weight of the concept, seen as a related concept to the concept c , $w_{old}(c_{related})$ is the old weight value of the concept, $sf_{c,c_{related}}$ is a semantic factor that depends of the type of semantic link existing between $c_{related}$ and c , and $w_{new}(c)$ is the new weight value for the current concept. It describes the influence (semantic effect) that concepts c has on concept $c_{related}$.

A special care has to be given to the definition of the semantic factor sf , which may decrease with the level of semantic proximity between $c_{related}$ and c :

- *Level 1.* $c_{related}$ is part of the definition of c . The relationships to be considered are of any type (transitive, inverse, etc.). One example is the direct concepts that constitute the definition of a given other concepts. For example, direct concepts that relates to “car” are “wheels”, “road”, etc.
- *Level 2.* $c_{related}$ is related to c by a combination of the same transitive relationship, and $c_{related} \cap c \neq Class$, meaning that they have a super-type in common.
- *Level n .* $c_{related}$ is related to c through the combination of the n same transitive relationship, and $c_{related} \cap c \neq Class$, meaning that they have a super-type in common.

If it is rather intuitive that this semantic factor depends of the semantic proximity of the two concepts, another issue concerns the relative values between two semantic factors sf_{c,c_1} and sf_{c,c_2} , i.e. the problem of assigning different weights to the semantic relations between the concepts of an ontology. Although this approach is still in an early stage, it is thought to be promising and could rely on the attempts that have been made to define a notion of similarity or distance between two concepts, such as in [11, 33], which propose an approach to similarity between concepts dealing with not only atomic concepts with Inclusion relation but also complex concepts with all kinds of semantic relations.

4 Contextual Personalisation

The shallowest consideration is sufficient to notice that human preferences are complex, variable and heterogeneous, and that not all preferences are relevant in every situation [35]. For instance, if a user is consistently looking for some contents in the Formula 1 domain, it would not make much sense that the system prioritizes some Formula 1 picture with a helicopter in the background just because the user happens to have a general interest for aircrafts. In other words, in the context of Formula 1, aircrafts are out of (or at least far from) context. Context is a difficult notion to grasp and capture in a software system, and the elements than can, and have been considered in the literature under the notion of context are manifold: user tasks and goals, computing platform, network conditions, social environment, physical environment, location, time, noise, external events, text around a word, visual context of a graphic region, to mention a few.

Complementarily to the ones mentioned, we propose a particular notion, for its tractability and usefulness in semantic content retrieval: that of semantic runtime context, which we define as the background themes under which user activities occur within a given unit of time. Using this notion, a finer, qualitative, context-sensitive activation of user preferences can be defined. Instead of a uniform level of personalisation, user interests related to the context are prioritized, discarding the preferences that are out of focus. The problems to be addressed include how to represent such context and determine it at runtime, and how the activation of user preferences should be related to it, predicting the drift of user interests over time.

4.1 Context Representation

Our approach is based on a concept-oriented context representation, and the definition of distance measures between context and preferences as the basis for the dynamic selection of relevant preferences [35].

A runtime context is represented (is approximated) in our approach as a set of weighted concepts from the domain ontology. This set is obtained by collecting the concepts that have been involved, directly or indirectly, in the interaction of the user (e.g. issued queries and accessed items) with the system during a retrieval session. The context is built in such a way that the importance of concepts fades away with time (number of user requests back when the concept was referenced) by a decay factor. This simulates a drift of concepts over time, and a general approach towards achieving this follows. Therefore, at each point t in time, context can be represented as a vector $C(t) \in [0, 1]^{|O|}$ of concept weights, where each $x \in O$ is assigned a weight $C_x(t) \in [0, 1]$. This context value may be interpreted as the probability that x is relevant for the current semantic context. Additionally, time is measured by the number of user requests within a session. In our approach, $C(t)$ is built as a cumulative combination of the concepts involved in successive user requests, in

such a way that the importance of concepts fades away with time. Right after each user's request, a request vector $Req(t) \in X_O$ is defined. In the next step, an initial context vector $C(t)$ is defined by combining the newly constructed request vector $Req(t)$ from the previous step with the context $C(t-1)$, where the context weights computed in the previous step are automatically reduced by the mentioned decay factor ξ , a real value in $[0,1]$. The decay factor will define how many *action units* will be considered for context-building, and how fast a concept will be "forgotten" by the system.

Once a context is built, the contextual activation of preferences is achieved by a computation of the semantic similarity between each user preference and the set of concepts in the context. In spirit, the approach consists of finding semantic paths linking preferences to context, where the considered paths are made of existing semantic relations between concepts in the domain ontology. The shorter, stronger, and more numerous such connecting paths, the more in context a preference is considered.

The proposed techniques to find these paths use a form of constrained spreading activation (CSA) strategy [13], similar to that explained in Sect. 3.2 for preference learning. In fact, in our approach a semantic expansion of both user preferences and the context takes place, during which the involved concepts are assigned preference weights and contextual weights, which decay as the expansion grows farther from the initial sets. This process can also be understood as finding a sort of fuzzy semantic intersection between user preferences and the semantic runtime context, where the final computed weight of each concepts represents the degree to which it belongs to each set.

Finally, the perceived effect of contextualisation is that user interests that are out of focus, under a given context, are disregarded, and only those that are in the semantic scope of the ongoing user activity (the "intersection" of user preferences and runtime context) are considered for personalisation. The inclusion or exclusion of preferences is in fact not binary, but ranges on a continuum scale, where the contextual weight of a preference decreases monotonically with the semantic distance between the preference and the context.

Contextualised preferences can be understood as an improved, more precise, dynamic, and reliable representation of user preferences, and as such they can be used directly for the personalised ranking of content items and search results, as described in Sect. 4.1, or they can be input to any system that exploits this information in other ways, such as the one described in Sect. 4.2.

4.2 Evaluation of Personalisation in Context

The contextualisation techniques described in this section have been implemented in an experimental prototype, and tested on a medium-scale corpus. Evaluating personalisation is known to be a difficult and expensive task [31, 36]. In order to measure how much better a retrieval system can perform with the proposed techniques than without them, it is necessary to

compare the performance of retrieval (a) without personalisation, (b) with simple personalisation, and (c) with contextual personalisation. The standard evaluation measures from the IR field require the availability of manual content ratings with respect to (i) query relevance, (ii) query relevance and general user preference (i.e. regardless of the task at hand), and (iii) query relevance and specific user preference (i.e. constrained to the context of his/her task).

For this purpose, we have conducted an experiment with real human subjects, focusing on the testing of the retrieval performance. The search space is based on a textual corpus, consisting of 145,316 documents (445 MB) from the CNN web site ([http://dmoz.org/News/Online Archives/CNN.com](http://dmoz.org/News/Online%20Archives/CNN.com)), plus the KIM domain ontology and KB [23], publicly available as part of the KIM Platform, developed by Ontotext Lab, with minor extensions. The Ontology Knowledge Base contains a total of 281 RDF classes, 138 properties, 35,689 in-stances, and 465,848 sentences. The CNN documents are annotated with KB concepts, amounting to over three million annotations in total. The user-neutral retrieval system used for this experiment is a semantic search engine developed by the authors [10]. Human subjects are given three different retrieval tasks, each expressing a specific information need, so that users are given the goal of finding as many documents as possible which fulfil the given needs. The sequence of actions is not fixed but is defined with full freedom by users as they seek to achieve the proposed tasks. A total of 18 subjects were selected for the experiment, all of them being PhD students from the authors' institutions. Three tasks were set up for the experiment, which can be briefly summarized as

1. News about agreements between companies.
2. Presentations of new electronic products.
3. Information about cities hosting a motor sports event.

Each task was tested (a) with contextual personalisation, (b) with simple personalisation, and (c) without personalisation. In order for users not to repeat the same task twice or more, each of the three modes was used with six users ($3 \text{ modes} \times 6 \text{ users} = 18 \text{ tests}$ for each task), in such a way that each user tried each of the three modes a, b, and c, exactly once. This way, each mode is tried exactly 18 times: once for each user, and six times for each task, in such a way that neither mode is harmed or favoured by different task difficulty or user skills. User preferences are obtained manually from the user by asking her to explicitly rate a predefined list of domain concepts at the beginning of the session. The relevant documents for each task are marked beforehand by an expert (a role that we played ourselves), so that users are relieved from providing extensive relevance judgements. However users are encouraged to open the documents that seem more relevant according to their subjective interests, in order to provide the system with more contextual tips. Context information is gathered based on concepts annotating such selected results, and the concepts that are related to the keywords in user queries (using the keyword-concept mapping provided in the KIM KB).

At the end of every task the system asks the user to mark the documents in the final result set as related or unrelated to her particular interests and the search task. For the computation of precision and recall after the experiment logs were collected, the following two simplifications are made for each interactive sequence (i.e. for each task and user):

- The search space is simplified to be the set of all documents that have been returned by the system at some point in the iterative retrieval process for the task conducted by this user.
- The set of relevant documents is taken to be the intersection of the documents in the search space marked as relevant for the task by the expert judgement, and the ones marked by the user according to her particular interests.

Figure 4 shows the results obtained with this setup and methodology. The curve on the left of this figure shows a clear improvement at high precision levels by the contextualisation technique both with respect to simple personalisation and no personalisation. The improvement decreases at higher recall levels. The cut-off precision curve clearly shows a significant performance improvement by the contextual personalisation, especially in the top 10 results. Personalisation alone achieves considerably lower precision on the top documents, showing that the contextualisation technique avoids further false positives which may still occur when user preferences are considered out of context. The mean average precision values for contextual, simple, and no personalisation in this experiment were 0.135, 0.106, and 0.046, respectively, which reflects that our technique globally performs clearly above the two baselines.

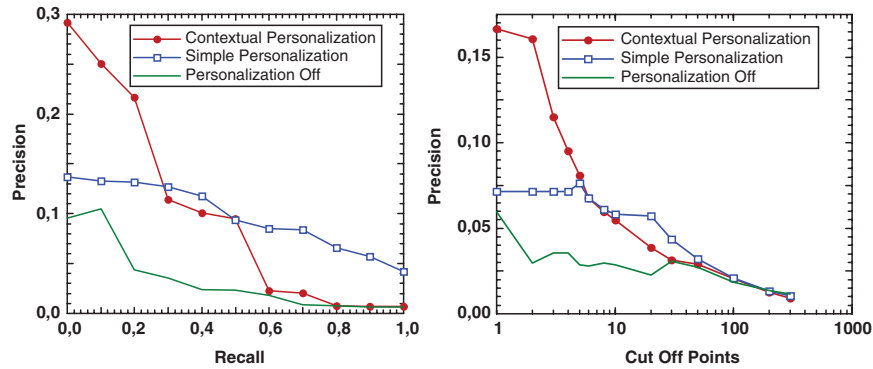


Fig. 4. Comparative performance of personalised search with and without contextualisation tested with 18 subjects on three proposed tasks. The graphics show (a) the precision vs. recall curve, and (b) the precision at cut-off points. The results are averaged over the set of all users and tasks

Most cases where our technique performed worse were due to a lack of information in the KB, as a result of which the system did not find that certain user preferences were indeed related to the context. Another limitation of our approach is that it assumes that consecutive user queries tend to be related, which does not hold when sudden changes of user focus occur. However, not only the general improvements pay off on average, but the potential performance decay in such cases disappears after two or three queries, since the weight of contextual concepts decreases exponentially as the user keeps interacting with the system.

5 Augmented Social Networking and Collaborative Filtering

When the system perspective is widened to take in contextual aspects of the user, it is often relevant to consider that in most cases the user does not work in isolation. Indeed, the proliferation of virtual communities, computer-supported social networks, and collective interaction (e.g. several users in front of a set-top box), call for further research on group modelling, opening new problems and complexities. A variety of group-based personalisation functionalities can be enabled by combining, comparing, or merging preferences from different users, where the expressive power and inference capabilities supported by ontology-based technologies can act as a fundamental piece towards higher levels of abstraction [7, 8].

5.1 Semantic Group Profiling

Group profiling can be understood under the explicit presence of a priori given user groups, or as an activity that involves the automatic detection of implicit links between users by the system, in order to put users in contact with each other, or to help them benefit from each other's experience. In the first view, collaborative applications may be required to adapt to groups of people who interact with the system. These groups may be quite heterogeneous, e.g. age, gender, intelligence and personality influence on the perception and demands on system outputs that each member of the groups may have. The question that arises is how the system can adapt itself to the group in such a way that each individual benefits from the results.

In [7] we have explored the combination of the ontology-based profiles defined in Sect. 2 to meet this purpose, on a per concept basis, following different strategies from social choice theory [27] for combining multiple individual preferences. In our approach, user profiles are merged to form a shared group profile, so that common content recommendations are generated according to this new profile (see Fig. 5).

With the combination of several profiles using the considered group modelling strategies we seek to establish how humans create an optimal ranked

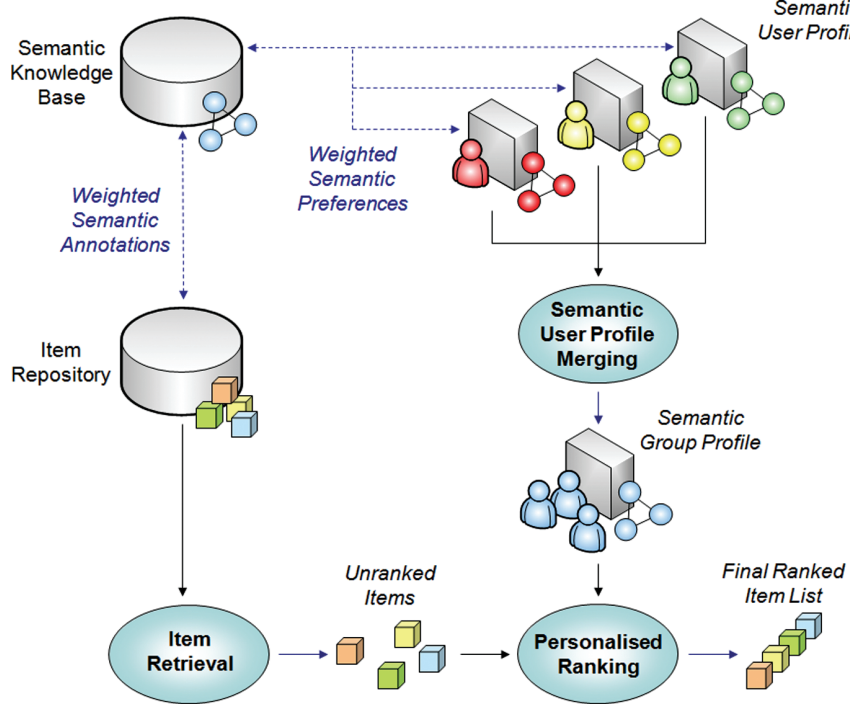


Fig. 5. Group profiling by aggregation of individual user profiles

item list for a group, and how they measure the satisfaction of a given list. Our preliminary experiments have shown that improved results can be obtained from the accuracy and expressivity of the ontology-based representation as proposed in this approach [7], and have exhibited which user profile combination strategies could be appropriate for a collaborative environment.

Specifically, we define a distance that measures the existing difference between two given ranked item lists, and we use this distance to determine which group modelling strategies give rankings 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 ranked item list for a given subject and let $\tau_{str} \in [0, 1]^N$ be the ranked item list for a specific combination strategy, where N is the number of items stored by the system. Using the notation $\tau(d)$ to refer the position of the item $d \in \mathcal{D}$ in the ranked list τ , we define the distance R between the previous ranked lists as follows:

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

where $P(n)$ is the probability that the user stops browsing the ranked item 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 distance R basically sums the differences between the positions of each item in the subject and strategy ranked lists. Thus, the smaller the distance is, the more similar the lists are. Additionally, and motivated by the fact that in typical information retrieval systems a user usually takes into account only the first n top ranked items, the formula considers more those items that appear before the n th position of the strategy ranking and after the n th position of the subject ranking. The idea here is to penalize those of the top n items in the strategy ranked list that are not relevant for the user.

The scenario of the experiments was the following. A set of 24 pictures was considered. For each picture several semantic-annotations were taken, 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 from the authors' institutions, and they were asked in all experiments 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*.

Observing the 24 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 the obtained lists we measured the distance R with respect to the ranked lists given by the group modelling strategies. 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 in the screen at first time after a query. Additionally, we also compared the strategies lists with those obtained using semantic user profiles in our personalised retrieval model explained in Sect. 2.

The average results are shown in Fig. 6. Surprisingly, both comparisons resulted quite similar. They agree with the strategies that seem to be more or less adequate for group modelling. From the figure, it can be seen that strategies like "Borda Count" and "Copeland Rule" give lists more similar to those manually created by the subjects, and strategies such as "Average Without Misery" and "Plurality Voting" obtained the greatest distances. For more details, see [7].

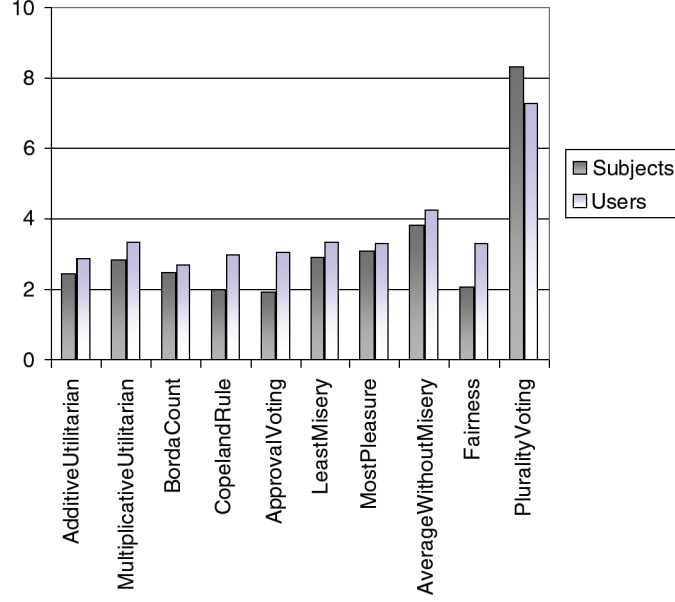


Fig. 6. Average distance R between the ranked lists obtained with the combination strategies, and the lists created by the subjects and the lists retrieved using the individual semantic user profiles

5.2 Semantic Social Networking

Even when explicit groups are not defined, users may take advantage of the experience of other users with common interests, without having to know each other. The issue of finding hidden links between users based on the similarity of their preferences or historic behaviour is not a new idea. In fact, this is the essence of the well-known collaborative recommender systems [3], where items are recommended to a certain user concerning those of his interests shared with other users or according to opinions, comparatives, and ratings of items given by similar users.

However, in typical approaches, the comparison between users and items is done globally, in such a way that partial, but strong and useful similarities may be missed. For instance, two people may have a highly coincident taste in cinema, but a very divergent one in sports. The opinions of these people on movies could be highly valuable for each other, but risk to be ignored by many collaborative recommender systems, because the global similarity between the users might be low.

In recommendation environments there is an underlying need to distinguish different layers within the interests and preferences of the users. Depending on the current context, only a specific subset of the segments (layers) of a user profile should be considered in order to establish his similarities with other people when a recommendation has to be performed. Models of

social networks partitioned into different common semantic layers can achieve more accurate and context-sensitive results.

The definition and generation of such models can be facilitated by a more accurate semantic description of user preferences, as supported by ontologies. A multilayered approach to social networking can be developed by dividing user profiles into clusters of cohesive interests, so that several layers of social networks are found. This provides a richer model of interpersonal links, which better represents the way people find common interests in real life.

Taking advantage of the relations between concepts, and the (weighted) preferences of users for the concepts, we have defined a strategy that clusters the semantic space. The obtained clusters are based on the correlation of concepts appearing in the preferences of individual users, representing thus those sets of preferences shared by specific cliques of people [8]. Considering the concept clusters, the user profiles are partitioned by projecting the clusters into the set of preferences of each user (see Fig. 7). Thus, users can be compared on the basis of the resulting subsets of interests, in such a way that several, rather than just one, (weighted) links can be found between two users.

Multilayered social networks are potentially useful for many purposes. For instance, users may share preferences, items, knowledge, and benefit from each other's experience in focused or specialized conceptual areas, even if they have very different profiles as a whole. Such semantic subareas need not be defined manually, as they emerge automatically with our proposed method. Users may be recommended items or direct contacts with other users for different aspects of day-to-day life.

In addition to these possibilities, our two-way space clustering, which finds clusters of users based on the clusters of concepts built in a first pass, offers a reinforced partition of the user space that can be exploited to generate group profiles for sets of related users as explained in Sect. 5.1. These group profiles enable efficient strategies for collaborative recommendation in real-time, by using the merged profiles as representatives of classes of users.

On the other hand, the degree of membership of the obtained sub-profiles to the clusters, and the similarities among them, can be used to define social

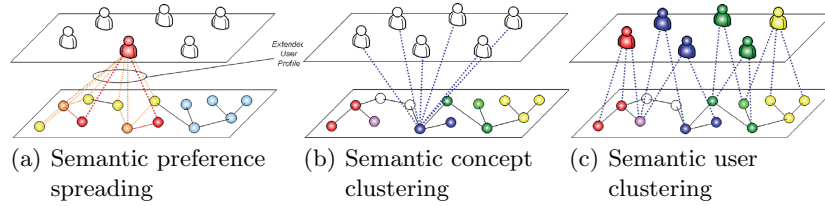


Fig. 7. Multilayer generation of social links between users: (a) the initial sets of individual interests are expanded, (b) domain concepts are clustered based on the vector space of user preferences, and (c) users are clustered in order to identify the closest class to each user

links to be exploited by collaborative filtering systems. We report early experiments with real subjects in [8], where the emergent augmented social networks are applied to a variety of collaborative filtering models, showing the feasibility of the clustering strategy.

Specifically, for a user profile $\mathbf{u}_m = (u_{m,1}, u_{m,2}, \dots, u_{m,K})$, an item vector $\mathbf{d}_n = (d_{n,1}, d_{n,2}, \dots, d_{n,K})$, and a cluster C_q , we denote by \mathbf{u}_m^q and \mathbf{d}_n^q the projections of the 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.

The profile of the user u_m is used to return a unique list. The score of an item d_n is computed as a weighted sum of the indirect preferences based on similarities with other users in each cluster. The sum is weighted by the similarities with the clusters:

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

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 normalized 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 normalized 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 .

This model, which we have called UP (user profile-based), can be slightly simplified if we only consider the similarities within the cluster C_q for which the user has the highest membership. With this simplification, we present the model UP- q (user profile-based) as follows:

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

If the current semantic cluster is well identified for a specific item, we expect to achieve better precision/recall results than those obtained with the overall model.

The experiments with real subjects we conducted in order to evaluate the effectiveness of our proposed social networking and recommender models

were very similar to those explained in Sect. 5.1. Again, the set of 24 pictures was defined as the retrieval space. Each picture was annotated with weighted semantic metadata describing what the image depicts, using an ontology including six topics: *animals*, *beach*, *construction*, *family*, *motor* and *vegetation*. The 20 Ph.D. students that participated in the previous experiments were asked to define their weighted preferences for a list of concepts related to the above topics, and present also in the annotations of the pictures. No restriction was imposed on the number of preferred topics and concepts to be selected by the subjects. The generated user profiles showed very different features, observable not only in their joint interests, but also in their complexity. Once the user profiles were created, we run our method. After the execution of the preference spreading procedure, the domain concept space was clustered according to interest similarity.

We evaluated the recommendation models UP and UP- q computing their average precision/recall curves for the users of each of the existing clusters. In this case we calculate the curves at different number of clusters ($Q = 4, 5, 6$). Figure 8 exposes the results.

The UP- q version, which returns ranked lists according to specific clusters, outperforms the UP version, which generates a unique list assembling the contributions of the users in all the clusters. Additionally, for both models, we have plotted with dotted lines the curves obtained without spreading preferences. It can be observed that our clustering strategy performs better when it is combined with the constrained spreading activation algorithm, thus showing preliminary evidence of the importance of extending the profiles before the clustering processes, as it is discussed in Sect. 5.3.

5.3 Semantic Profile Expansion for Collaborative Group Profiling

In real scenarios, user profiles tend to be very scattered, especially in those applications where user profiles have to be manually defined. Users are usually not willing to spend time describing their detailed preferences to the system, even less to assign weights to them, especially if they do not have a clear understanding of the effects and results of this input. On the other hand, applications where an automatic preference learning algorithm is applied tend to recognize the main characteristics of user preferences, thus yielding profiles that may entail a lack of expressivity. To overcome this problem, the semantic preference spreading mechanism described in Sect. 3 has proved highly useful for improving our group profiling techniques as well.

Previous experiments without the semantic spreading feature showed considerably poorer results. The profiles were very simple and the matching between the preferences of different users was low. Typically, the basic user profiles provide a good representative sample of user preferences, but do not reflect the real extent of user interests, which results in low overlaps between the preferences of different users. Therefore, the extension is not only

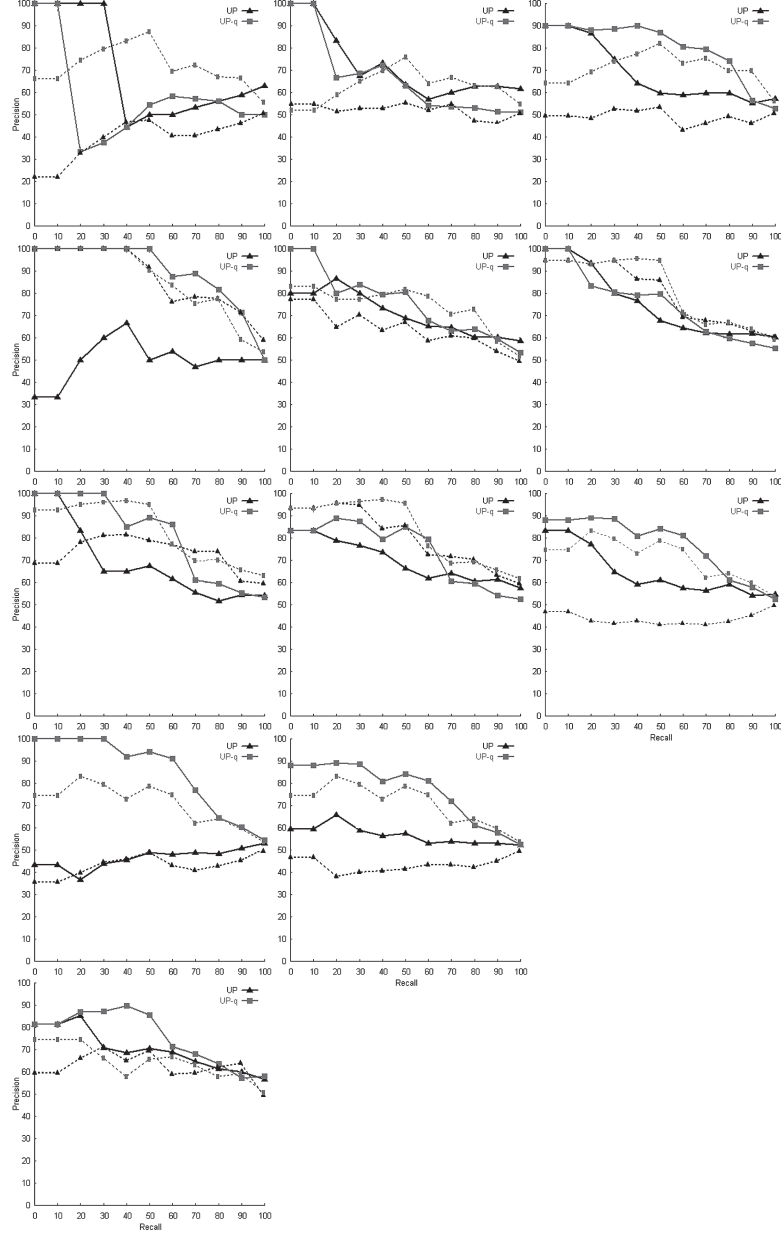


Fig. 8. Average precision vs. recall curves for users assigned to the user clusters obtained with the UP (black lines) and UP-q (gray lines) models at levels of $Q = 6$ (graphics on the left), $Q = 5$ (graphics in the middle), and $Q = 4$ (graphics on the right) concept clusters. For both models, the dotted lines represent the results achieved without semantic preference spreading

important for the performance of individual personalisation, but is essential for the clustering strategy described in Sect. 5.2.

In very open collaborative environments, it is also the case that not only direct evidence of user interests needs to be properly completed in their semantic context, but that they are not directly comparable with the input from other users in its initial form. If the environment is very heterogeneous, the potential disparity of vocabularies and syntax used by different users or subsystems pose an additional barrier for collaborative techniques. One of the major purposes for which ontologies are conceived is that of reflecting or achieving a consensus between different parties in a common knowledge space [17]. Therefore, they provide special-purpose facilities to ensure the required interoperability between semantic user spaces, and match descriptions that are syntactically different but semantically related.

6 Conclusions

Ontology-based technology has been identified as a key tool for moving beyond current state of the art in knowledge-intensive fields, where a precise, formal representation of the involved semantics can make a difference. In this chapter we address the potential of the ontology-based approach in the area of personalisation for content access and retrieval. We have shown ways in which ontology-driven representations can be used to improve the effectiveness of different personalisation techniques, focusing on specific functionalities where the introduction of ontologies enables qualitative enhancements. In the proposed approaches, ontologies are used to model the domain of discourse in terms of which user interests, content meaning, retrieval context, and social relationships, can be described, analysed, and interrelated.

The presented research is being currently continued in the MESH⁵ project, in which the proposed ontology-driven mechanisms are combined to provide users with personalised multimedia news retrieval. Automatic personalisation has a particularly high potential in the news domain, because personal preferences play an important role in the way users approach, move and browse across the space of available information. Indeed, there are many situations where users seek information for the sake of it, without the necessary intervention of a specific, immediate goal or task beyond their own personal interest or whim. As such, this project provides a good recipient for further experimenting with the benefits of ontology-driven personalisation, and concurring with the development of a general personalisation infrastructure, supporting user profile construction and evolution, and contextual, semantic preference-based content retrieval functionalities.

The advantages of ontology-driven representations (expressiveness and precision, formal properties, inference capabilities, interoperability) enable

⁵ Multimedia sEmantic Syndication for enHanced news services, <http://www.mesh-ip.eu>

further developments that exploit such capabilities, beyond the ones proposed here, on top of the basic personalisation framework described in this chapter. A trade-off of our proposals is the cost and difficulty of building well-defined ontologies and populating large-scale knowledge bases, which is not addressed here. Recent research on these areas is yielding promising results [23], in a way that any advancement on these problems can be played to the benefit of our proposed achievements.

Acknowledgements

The research leading to this document has received funding from the European Community's Sixth Framework (FP6-027685-MESH), and the Spanish Ministry of Science and Education (TIN2005-06885). However, it reflects only the authors' views, and the European Community is not liable for any use that may be made of the information contained therein.

References

1. Badros, G.J. and Lawrence, S.R. (2005). Methods and systems for personalised network searching. *US Patent Application 20050131866*.
2. Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley, Reading, MA.
3. Balabanovic, M. and Shoham, Y. (1997). Content-based collaborative recommendation. *Communications of the ACM* 40(3), pp. 66–72.
4. Bloehdorn, S., Petridis, K., Saathoff, C., Simou, N., Tzouvaras, V., Avrithis, Y., Handschuh, S., Kompatsiaris, Y., Staab, S., and Strintzis, M. G. (2005). Semantic annotation of images and videos for multimedia. *Proceedings of the Second European Semantic Web Conference (ESWC 2005)*, Heraklion, Greece. Springer Verlag Lecture Notes in Computer Science, vol. 3532, pp. 592–607.
5. Block, J. and Funder, D.C. (1986). Social roles and social perception: Individual differences in attribution and “error”. *Journal of Personality and Social Psychology* 51, pp. 1200–1207.
6. Bookstein, A. (1983). Outline of a general probabilistic retrieval model. *Journal of Documentation* 39(2), pp. 63–72.
7. Cantador, I., Castells, P., and Vallet, D. (2006). Enriching group profiles with ontologies for knowledge-driven collaborative content retrieval. *Proceedings of the First International Workshop on Semantic Technologies in Collaborative Applications (STICA 2006)*, at the 15th IEEE International Workshops on Enabling Technologies (WETICE 2006), Manchester, UK, pp. 358–363.
8. Cantador, I. and Castells, P. (2006). Multilayered semantic social network modelling by ontology-based user profiles clustering: application to collaborative filtering. *Proceedings of the 15th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2006)*, Podesbrady, Czech Republic. Springer Verlag Lectures Notes in Artificial Intelligence, vol. 4248, pp. 334–349.

9. Castells, P., Fernández, M., Vallet, D., Mylonas, P., and Avrithis, Y. (2005). Self-tuning personalised information retrieval in an ontology-based framework. *Proceedings of the First International Workshop on Web Semantics (SWWS 2005)*, Agia Napa, Cyprus. Springer Verlag Lecture Notes in Computer Science, vol. 3762, pp. 977–986.
10. Castells, P., Fernández, M., and 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.
11. Cross, V. (2004). Fuzzy semantic distance measures between ontological concepts. *IEEE Annual Meeting of the Fuzzy Information*, pp. 635–640.
12. Darragh, J.J., Witten, I.H., and James, M.L. (1990). The reactive keyboard: a predictive typing aid. *IEEE Computer* 23(11), pp. 41–49.
13. Crestani, F. (1997). Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review* 11, pp. 453–482.
14. Gauch, S., Chaffee, J., and Pretschner, A. (2003). Ontology-based personalised search and browsing. *Web Intelligence and Agent Systems Journal* 1(3–4), pp. 219–234.
15. Gauch, S. and Trajkova, J. (2004). Improving ontology-based user profiles, *Proceedings of the Recherche d’Informations Assistée par Ordinateur (RAIO 2004)*, Avignon, France, April 2004, pp. 380–389.
16. Good, N., Schafer, J., Konstan, J., Borchers, A., Sarwar, B., Herlocker, J., and Riedl, J. (1999). Combining collaborative filtering with personal agents for better recommendations. *Proceedings of the 16th National Conference on Artificial Intelligence (AAAI 1999)*, Orlando, FL, pp. 439–446.
17. Gruber, T.R. (1993). A translation approach to portable ontology specification. *Knowledge Acquisition* 5, pp. 199–220.
18. Hanani, U., Shapira, B., and Shoval, P. (2001). Information filtering: overview of issues, research and systems. *User Modeling and User-Adapted Interaction* 11(3), pp. 203–259.
19. Hirsh, H., Basu, C., and Davison, B.D. (2000). Enabling technologies: learning to personalize. *Communications of the ACM* 43(8), pp. 102–106.
20. Hull, D.A. (1998). The TREC-7 filtering track: description and analysis. *Proceedings of the Seventh International Text REtrieval Conference (TREC-7)*, Gaithersburg, Maryland, pp. 33–56.
21. Jeh, G. and Widom, J. (2003). Scaling personalised web search. *Proceedings of the 12th International World Wide Web Conference (WWW 2003)*, Budapest, Hungary, pp. 271–279.
22. Kim, H. R. and Chan, P.K. (2003). Learning implicit user interest hierarchy for context in personalisation. *Proceedings of the Eighth International Conference on Intelligent User Interfaces (IUI 2003)*, Miami, FL, pp. 101–108.
23. Kiryakov, A., Popov, B., Terziev, I., Manov, D., and Ognyanoff, D. (2004). Semantic annotation, indexing, and retrieval. *Journal of Web Semantics* 2(1), pp. 47–49.
24. Koychev, I. and Schwab, I. (2000). Adaptation to Drifting user’s interests. *Proceedings of the Workshop on Machine Learning in New Information Age (MLNet 2000) at the 11th European Conference on Machine Learning (ECML 2000)*, Barcelona, Spain, pp. 39–45.
25. Koychev, I. and Lothian, R. (2005). Tracking drifting concepts by time window optimisation. *Proceedings of AI-2005, the 25th SGAI International Conference*

- on Innovative Techniques and Applications of Artificial Intelligence*. Bramer, Max; Coenen, Frans; Allen, Tony (Eds.), Springer, London, pp. 46–59.
26. Linden, G., Jacobi, J., and Benson, E. (2001). Collaborative recommendations using item-to-item similarity mappings, *US Patent 6266649B1*.
27. 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.
28. Micarelli, A. and Sciarone, F. (2004). Anatomy and empirical evaluation of an adaptive web-based information filtering system. *User Modelling and User-Adapted Interaction* 14(2–3), pp. 159–200.
29. Padmanabhan, V.N. and Mogul, J.C. (1996). Using predictive prefetching to improve World Wide Web latency. *Computer Community Review* 26(3), pp. 22–36.
30. Papadogiorgaki, M., Papastathis, V., Nidelkou, E., Waddington, S., Bratu, B., Ribière, M., and Kompatsiaris, Y. (2007). Distributed user profile management and adaptation for personalised news content delivery, submitted to the special issue “Data Mining for Personalisation” of *User Modelling and User Adapted Interaction* (UMUAI) Journal.
31. Rajagopalan, B. and Deshmukh, A. (2005). Evaluation of online personalisation systems: a survey of evaluation schemes and a knowledge-based approach. *Journal of Electronic Commerce Research* 6(2), pp. 112–122.
32. Smith B.R., Linden, G.D., and Zada, N.K. (2005). Content personalisation based on actions performed during a current browsing session, *US Patent 6853983B2*.
33. Song, L., Ma, J., Li, L., and Chen, Z. (2006). Fuzzy similarity from conceptual relations. *IEEE Asia-Pacific Conference on Services Computing* (APSCC 2006), pp. 3–10.
34. Tsymbal, A. (2004). The problem of concept drift: definitions and related work, *Technical Report TCD-CS-2004-15*, Department of Computer Science, Trinity College Dublin, Ireland.
35. Vallet, D., Castells, P., Fernández, M., Mylonas, P., and Avrithis, Y. (2007). Personalised content retrieval in context using ontological knowledge. *IEEE Transactions on Circuits and Systems for Video Technology* 17(3), *Special Issue on the Convergence of Knowledge Engineering Semantics and Signal Processing in Audiovisual Information Retrieval*, pp. 336–346.
36. White, R.W., Jose, J.M., van Rijsbergen, C.J., and Ruthven, I. (2004). A simulated study of implicit feedback models. *Proceedings of the 26th European Conference on Information Retrieval* (ECIR 2004), Sunderland, UK, pp. 311–326.
37. Zamir, O.E., Korn, J.L., Fikes, A.B., and Lawrence, S.R. (2005). Personalisation of placed content ordering in search results. *US Patent Application 20050240580*.

Advances in Semantic Media Adaptation and
Personalization

Wallace, M.; Angelides, M.C.; Mylonas, P. (Eds.)

2008, XIV, 368 p., Hardcover

ISBN: 978-3-540-76359-8