Peersommender: A Peer-level Annotation-based Approach for Multimedia Recommendation

Marcelo G. Manzato and Rudinei Goularte

Instituto de Ciências Matemáticas e de Computação – ICMC-USP
Universidade de São Paulo, Brazil
{mmanzato, rudinei}@icmc.usp.br

Abstract. In this article, we propose the Peersommender architecture, which is the set of applications that provide personalized content to users according to their annotations produced when watching multimedia items. As opposite to hierarchical authoring, which is metadata created by experts to describe content in an organized, structured and impartial manner, peer-level annotations are highly personal because they are created by consumers, and this feature can be used to infer relevant content that is of interest to the user. Particularly, we propose a movie recommender system that explores a user profile with automatic augmentation, which is based on annotations produced by the user in the past. By combining tags, faces of interest and ratings with usual hierarchical metadata, we are able to predict ratings for new movies based on an enhanced hybrid approach for content filtering. Our evaluation was executed over a large scale dataset containing real users, and it shows good results when compared to other techniques.

Categories and Subject Descriptors: H. Information Systems [**H.m. Miscellaneous**]: Databases

Keywords: Collaborative and content-based filtering, peer-level annotation, profiling, recommendation, semantic information

1. INTRODUCTION

Due to the increasing amount of multimedia content that is available each day, the provision of personalization services has become crucial in order to help users to deal with information overload. As a particular type of personalization, recommender systems make available a selection of items and services, which are chosen in a way to match the user's preferences and interests [Adomavicius and Tuzhilin 2005]. Usually, the process of gathering user's personal information and further selection of items is accomplished by following three different strategies: content-based recommendations, collaborative recommendations and hybrid approaches [Balabanovic and Shoham 1997].

In spite of the variety of ways to recommend multimedia items, the achievements are highly dependent on the knowledge about the users' personal information, and also, the viability of semantic metadata describing the content itself. In the first case, systems such as last.fm[1], Netflix[2] and Youtube[3] usually build user profiles containing her preferences based on the history of visited items, and also, according to feedback given manually by the user, such as assigning ratings or filling forms with personal data. Augmenting user profiles, however, is limited to these user activities, not considering other possibilities to gather personal interests, which, in fact, could benefit the recommending process by providing more information about the user.

[1]http://www.last.fm/
[2]http://www.netflix.com/
[3]http://www.youtube.com/

In the second case, the description of multimedia content is typically provided in the content provider side by professionals, who create well-structured information about specific media items, helping search and analysis tasks. Due to the taxonomy-style and organization of those metadata, some authors named those descriptions as hierarchical authoring [Bulterman 2004]. Such hierarchical information can be extracted automatically (without human intervention) or manually (with user intervention). Automatic approaches are usually highly dependent on the item's domain; whereas in manual approaches, the description is considered a time-consuming task and error prone [Patel and Abowd 2004], which requires huge effort from producers to accomplish metadata creation for the volume of multimedia data available nowadays.

As an attempt to minimize the aforementioned problems about user profiling and semantic content description, advances in the Web 2.0 field have allowed the exploration of user-generated annotations in order to obtain metadata about the content, and also, information about user's interests. In this environment, services such as Youtube, Facebook[4] and Flickr[5] encourage users to act as producers, being able to author new videos, photos, audio and documents, and make them available on the Web. More important, it is possible to annotate, augment and/or enrich existent data, by inserting tags, comments, handwritten notes, and also, links to other related media. Usually, those user-generated annotations, named peer-level annotations [Bulterman 2004], may contain concrete and rich information about user's preferences, which is valuable for personalization services. Moreover, they do not follow a restrictive vocabulary, and can be created using different interaction paradigms.

In previous work [Manzato et al. 2009; Manzato and Goularte 2009], we started our research to support user's interests discovery and metadata extraction by exploring peer-level annotations. As a result, those techniques are able to recommend related scenes to the user according to her preferences, which are obtained from her interaction with the current video being watched. However, a number of issues still have to be addressed: i) the proposal of a general architecture able to handle personalization applications in the Web 2.0 environment; ii) the formal description of how peer-level annotations, hierarchical metadata and users' interests relatedness can be used together in order to create an augmented user profile and, consequently, improve recommendations; and iii) what is the real impact of using user-generated metadata in personalization applications. In this article, we explore these issues by describing a general architecture called Peersommender (Peer-level annotation based recommender system). It consists of providing personalization services in the Web 2.0 environment, considering annotations produced by users to infer their personal interests. Particularly, we propose a movie recommender system that considers tags, ratings and faces of interest to improve known content and collaborative-based filtering algorithms. We evaluate our strategy by comparing our results to other approaches previously proposed in the literature.

This article is structured in a way to present all involved modules of the Peersommender architecture. After the related work about recommender systems, which is described in Section 2, the general overview of the system is depicted in Section 3. After that, the hierarchical content description and the peer-level annotation procedures are presented in Section 4. Using those metadata, we propose in Section 5 the user profiling mechanism, which is one of the contributions of this article. Based on this user profile, a set of recommender algorithms is described in Section 6, including one based on peer-level annotations, which is another contribution of this article. Section 7 depicts the results obtained with our approach, by comparing it with a set of well-known recommender algorithms; and finally, in Section 8, the conclusions and future work are presented.

[4]http://www.facebook.com/
[5]http://www.flickr.com/

2. RELATED WORK

The related work about recommender systems, in general, adopt one of the following approaches: i) content-based recommendations [Pazzani and Billsus 1997], where the user will be recommended items similar to the ones he/she accessed and liked in the past; ii) collaborative recommendations [Breese et al. 1998; Cohen et al. 1999], where content selection is based on similar interests from other people; and iii) hybrid approaches [Pazzani 1999], which combine collaborative and content-based recommendations.

Regarding content-based recommenders, the selection of items is based on a user profile built from previously watched movies' metadata, such as title, keywords, genres, etc. [Gauch et al. 2007]. The user feedback (e.g., ratings) is used to calculate weights for concepts of interest to the user, which are further explored by the system to search for related content. In this searching procedure, a variety of information retrieval techniques can be used [Pazzani and Billsus 1997], being, most of them, based on textual information. One very-known problem of content-based filtering is overspecialization, which occurs when the system can only recommend items that score highly against the user profile, resulting in a limitation of the user to being recommended items that are similar to those already rated [Adomavicius and Tuzhilin 2005].

In order to minimize the problem of overspecialization, collaborative filtering can be used, as it considers the similarity of users or items: in the first case, items liked by a group of people with similar interests are recommended to a particular user from that group; in the second, users' ratings are predicted for an item based on their ratings for similar items [Resnick et al. 1994; Koren et al. 2009]. The overspecialization is overcome because, different from content-based, the system does not know any metadata about the content, and thus, the algorithms rely only on ratings previously assigned by other users. Maybe the most remarkable effort spent on collaborative recommenders was during the Netflix Prize[6], which contributed to the emergence of two new trends of research: the use of singular value decomposition (SVD) to cluster similar movies and users using a variety of latent semantic factors [Koren et al. 2009], and the combination of multiple recommender algorithms to improve performance [Bell and Koren 2007]. Although collaborative filtering can minimize overspecialization, it still has some limitations, such as the new user and new item problems, or when a user has distinct preferences from all other users.

The use of hybrid approaches, in turn, can surpass most problems inherent to content-based and collaborative filtering. It combines both methods in one of the following ways: i) implementing collaborative and content-based algorithms separately and combining their predictions linearly [Claypool et al. 1999; Pazzani 1999]; ii) incorporating some content-based characteristics into a collaborative approach [Balabanovic and Shoham 1997; Pazzani 1999]; iii) incorporating some collaborative characteristics into a content-based approach [Soboroff and Nicholas 1999]; and iv) constructing a general unifying model that incorporates both content-based and collaborative characteristics [Popescul et al. 2001; Schein et al. 2002]. Albeit these methods can predict better ratings, the process of extracting metadata from the content and gathering information from user's interests are issues that still need further research.

In the first case, a number of solutions have been proposed to extract semantic information from audiovisual data by processing low-level features, such as color information, motion vectors and histograms. Venkatesh et al. [Venkatesh et al. 2008] report some ways to automatically extract this high-level metadata; however, most techniques have good performance only in specific video domains. Manual approaches, in turn, such as the Music Genome Project[7] and the Internet Movie Database (IMDB)[8], have to deal with the time-consuming efforts to annotate lots of content that are available

[6]http://www.netflixprize.com/
[7]http://www.pandora.com/
[8]http://www.imdb.com/

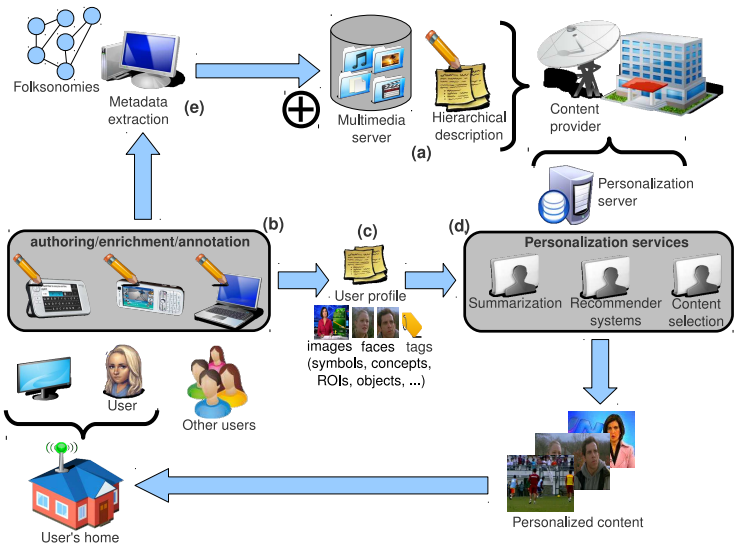

Fig. 1. Peersommender architecture.

each day [Patel and Abowd 2004].

In the second case, user profiling mechanisms usually start with personal information collection, which includes implicit/explicit user feedback, and the use of proxy servers, browser/desktop agents, among others. Having the necessary information, the gathered data is used to build a profile according to a number of strategies such as those based on keywords, semantic network profiles, concepts profiles, etc. [Gauch et al. 2007]. Nevertheless, all these techniques are limited to the amount of metadata available on the videos watched by the user, which means that recommendations cannot reach a semantic level of concepts of interest to the user.

On the other hand, Web 2.0 services have opened new possibilities to deal with the problem of metadata extraction, and also, the provision of richer information about the user's preferences. Concerning metadata extraction, notorious work can be found on collaborative tagging field [Angeletou et al. 2009; Heckner et al. 2008; Halpin et al. 2007], where the viability of tags assigned collaboratively to each resource can be explored in order to create folksonomies based on co-occurrence of tags. Concerning user profiling and recommendations, some systems are able to provide recommendations based on user-generated tags [Gemmis et al. 2008; Sen et al. 2009; Tso-Sutter et al. 2008; Zanardi and Capra 2008], assuming that when a user tags a video, that means this piece of content can bring relevant information about her preferences. However, most of those systems still lack research in the formal description of annotation-based profiling mechanisms, and their integration with folksonomies, in order to obtain better recommendations with the knowledge of associated semantic information.

Our work differs from the aforementioned systems because we deal not only with tags as user input to infer her preferences. As previously described, content enrichment and augmentation can be accomplished by using different interaction paradigms, and thus, we also consider assigned ratings to past items and faces of interest defined from handwritten strokes. In addition, the literature does not report a conceptual architecture and user profiling mechanisms that can support, in general terms, the development of personalization services in the scenario depicted so far in this article.

3. PROPOSED ARCHITECTURE

In this section, we describe the overall schema of the Peersommender architecture proposed in this article. Figure 1 illustrates the adopted scenario, where a user watching multimedia content is able

to enrich/annotate content using a variety of devices and interaction mechanisms (Figure 1 (b)). It is important to note that this possibility is a common feature typically provided by services in the Web 2.0 environment, such as Youtube, Flickr, etc. Annotations produced collaboratively by different users are analyzed in order to extract semantic metadata, as usually described by research in folksonomies creation field (Figure 1 (e)). This metadata is extra material for hierarchical descriptions, helping experts during the effort spent in content description (Figure 1 (a)). In parallel, personal descriptions produced by the user can be used to build a profile containing her personal preferences for subjects, concepts and content (Figure 1 (c)). Examples of this kind of information are: assigned ratings to watched movies, inserted tags, images, regions and faces of interest, and so forth. When combining personal information with content descriptions, the content provider is able to provide better personalization services to the user (Figure 1 (d)). Recommender systems and content selection tools, for instance, will offer the user movies or scenes which have higher probability to match her preferences. Summarization systems, in turn, will create a summarized version of the content containing only pieces of information which are relevant to the user.

In this work, we scope our study to recommender systems; however, such architecture can also be suited to other personalization services that explore user-generated data. In the next sections, each one of the architecture's components is depicted in details. The module illustrated by Figure 1 (e) is left to future work (although we still use folksonomies to augment user profiles).

4. MULTIMEDIA INDEXING AND ANNOTATION

This section depicts the modules delimited by Figure 1 (a) and (b). First, we present the hierarchical description adopted in this work; it consists of defining the types of metadata we are considering. In what follows, we describe the user enrichment activity by presenting the M4Note tool, which was previously reported elsewhere [Goularte et al. 2004].

4.1 Hierarchical Description

Hierarchical descriptions provide information about specific media items with the objective to be searched or analyzed [Bulterman 2004]. This type of information usually follows an organized, well-structured taxonomy which makes easy the access by content personalization applications. One example is metadata referring to a movie, such as title, producer, characters, etc. Such descriptions are usually provided by content providers, where experts are responsible to create information by carefully analyzing different factors of the content. In this article, we use a subset of information available on the Internet Movie Database (IMDB): list of genres of movies, and associated keywords.

It is important to note that further information available on IMDB could be explored, such as producer, list of characters, director, etc. However, the more descriptions are used in the system, the more efforts should be needed to annotate new videos available each day. In addition, our proposal is to use consumer-generated data in order to minimize this time-consuming effort. Thus, we have delimited the use of hierarchical descriptions to only a few of them. As further explained, all this considered metadata will be combined with user annotations, in order to make possible our movies recommender system.

4.2 User Annotation

The previous subsection described our adopted video description procedure, which is executed with all video streams stored into the multimedia database; it makes available the search of movies according to their content. This subsection, in turn, depicts the interaction environment which enables users to enrich the content.

The enrichment activity explores different interaction paradigms. A user can capture a frame, and

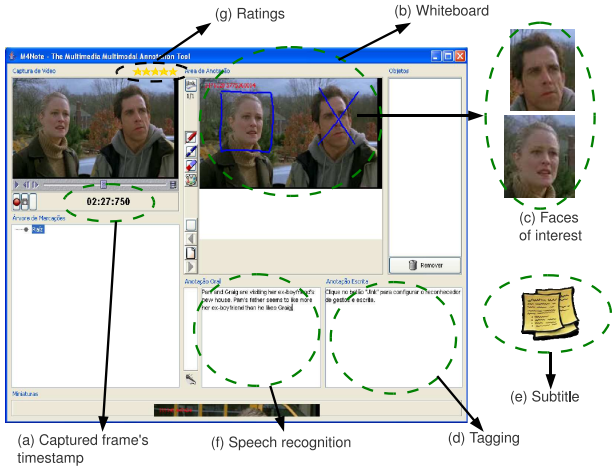

Fig. 2. Peer-level enrichment environment using the M4Note application. Dashed ellipses represent extracted metadata that describes user's preferences.

make handwriting strokes on selected frames, as well as assign ratings and tags, and make spoken commentaries. Figure 2 illustrates the M4Note tool [Goularte et al. 2004]. It is based on multimodal interfaces with electronic ink and voice, providing the user with annotation mechanisms over a multimedia object stream. This tool allows the user to accomplish the following types of annotation:

—Capture a frame and use it as an image background in the whiteboard area (Figure 2 (a)).
—Make handwritten notes using electronic ink onto the captured frame; the tool also recognizes known symbols like squares, circles, etc. (Figure 2 (b)).
—From the strokes produced by the electronic ink, faces of interest can be extracted using the horizontal and vertical maximum and minimum coordinates of each symbol made by the user (Figure 2 (c)), and further recognition using one of the following approaches: automatically, using automatic recognition over samples in database; or from user interaction, when the recognition module does not have enough samples of that face. In both cases, we use the Face Annotation Interface Java API (faint)[9], that extracts faces and stores them as thumbnails, together with the person's name.
—Provide textual information using tags (Figure 2 (d)), recognized speech (Figure 2 (e)), and/or subtitle of the video, which is delimited by the captured frame's timestamp.
—Assign rates to the video being watched (Figure 2 (g)). Such ratings vary from 0.5 (hated) to 5 (loved), with 0.5 of increment.

In this work, we focus our study on a subset of annotations and feedback provided by the user: tags, faces of interest and assigned ratings. When those metadata are combined together and with hierarchical descriptions, it is possible to create a user profile containing her major preferences. Next section, in turn, depicts how this process is accomplished.

5. USER PROFILING

This section presents in details the profiling module delimited by Figure 1 (c). Creating users profiles consists of storing their past activities in order to help personalization services to know what are the interests, likes and dislikes of each user. Some work in this field can be found in the literature, as described in [Gauch et al. 2007]; however, none of them considers metadata produced by the user him/herself.

[9]http://faint.sourceforge.net/

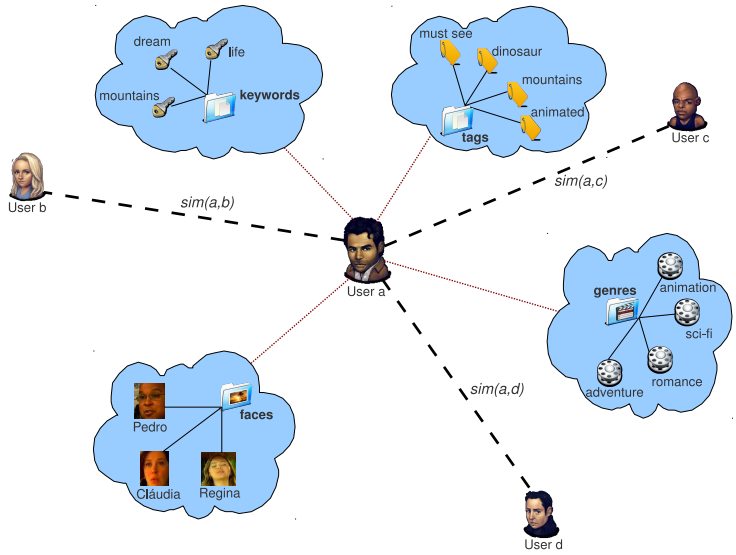

Fig. 3. User profile containing different types of metadata: tags, movies' genres, faces of interest, images and relation to other users with similar interests.

Our profiling approach deals with the notion of data clouds with interconnections among users with similar interests. Data clouds cluster a variety of types of information obtained from the different interaction paradigms between users and content. Figure 3 illustrates such profile, where four types of data clouds are attached to the main user (User "a" in Figure 3): keyword cloud, genre cloud, tag cloud and face cloud. We also provide an automatic augmentation procedure to the last two clouds, by considering co-occurrence of tags and faces produced by other users. This feature has the advantage of producing semantically richer information about user's preferences, minimizing the lack of data in cases where a subset of users do not provide much annotations.

In addition to data clouds, the user profile also contains a list of other users who have similar interests to the current one. The interconnections are created based on the rating values of common items watched in the past by two or more users. The relationship among users can be used in collaborative filtering algorithms, with the advantage of minimizing overspecialization problems, which are inherent to content-based filtering approaches [Adomavicius and Tuzhilin 2005].

Next subsections present in details the formal definition of each data cloud, plus the adopted users similarity function. Before doing that, however, we present in the next subsection, the notation adopted in this article [Szomszor et al. 2007].

5.1 Notation

Let us denote a given user by $u \in U$, where U is the set of all users, a movie item by $s \in S$, where S is the set of all available items, and a rating value by $r \in \{0.5, 1, 1.5, ..., 5\} \equiv R$. A set of items rated by user u is represented by S_u; and based on this set, we define the rating function for user u as $\delta_u : s \in S_u \mapsto \delta_u(s) \in R$.

Associated to data clouds, we also consider the following notations:

—We denote by K the global set of keywords, K_s the set of keywords associated to movie s, and N_k the global frequency of occurrence of keyword k for all movies.

—Analogously, we denote by G the global set of genres, G_s the set of genres associated to movie s, and N_g the global frequency of occurrence of genre g for all movies.

—Related to tags, we define T as the global set of tags, T_s the set of tags associated to movie s, T_u the set of tags created by user u, T_{us} the set of tags user u has associated to movie s, and N_t the global frequency of occurrence of tag t for all movies.

—The same for faces, F is the global set of recognized faces, F_s is the set of faces recognized from movie s, F_u is the set of faces scribbled and/or recognized by user u, F_{us} is the set of faces user u has scribbled/recognized from movie s, and N_f is the global frequency of occurrence of face f for all movies.

Given the adopted notation, we are able to formally define each data cloud in the following subsections; the last one, in particular, provides the definition of our user similarity function.

5.2 Keyword Cloud

A keyword cloud corresponds to a set of keywords describing the content obtained from hierarchical descriptions. Differently from tags, which can be any word or set of words provided by the user, keywords usually have significant meaning, and provide content metadata without being personal.

We define $cloud_k(u, r)$ as the **rating keyword cloud** for a given user u and rating r, containing the set of pairs $(k, n_{k,u,r})$, where $k \in K$ is a keyword and $n_{k,u,r} = |\{s \in S_u | k \in K_s \ \ \& \ \ \delta_u(s) = r\}|$. Thus, $n_{k,u,r}$ contains the frequency of occurrence of keyword k for all movies that user u has associated with rating r.

5.3 Genre Cloud

A genre cloud corresponds to a set of preferred genres of movies watched so far by the user. As depicted in last subsection, the genres are provided by hierarchical descriptions obtained from the IMDB archive.

We define $cloud_g(u, r)$ as the **rating genre cloud** for a given user u and rating r, containing the set of pairs $(g, n_{g,u,r})$, where $g \in G$ is a genre and $n_{g,u,r} = |\{s \in S_u | g \in G_s \ \ \& \ \ \delta_u(s) = r\}|$. Thus, $n_{g,u,r}$ contains the frequency of occurrence of genre g for all movies that user u has associated with rating r.

5.4 Tag Cloud

A tag cloud corresponds to all tags a user assigns to the items visited in the past, and their meaning can be used to infer the degree of interest he/she has about a set of concepts.

We define $cloud_t(u, r)$ as the **rating tag cloud** for a given user u and rating r, containing the set of pairs $(t_u, n_{t_u,u,r})$, where $t_u \in T$ is a tag created by user u and $n_{t_u,u,r} = |\{s \in S_u | t_u \in T_{us} \ \ \& \ \ \delta_u(s) = r\}|$. Thus, $n_{t_u,u,r}$ contains the frequency of occurrence of tag t_u for all movies that user u has associated with rating r.

In addition to tags created by the current user, we augment the tag cloud by considering their co-occurrence with tags provided by other users. This feature has the advantage of building a user profile with better semantic information, and also, minimizing the lack of data in cases where a subset of users does not provide many annotations.

Our tag cloud augmentation is based on the relatedness measure described in [Cattuto et al. 2008], who create a folksonomy based on co-occurrence of tags. They define it as a weighted undirected graph whose set of vertices is the set T of tags. Two tags t_1 and t_2 are connected by an edge if and only if $t_1, t_2 \in T_{us}$; and its weight is given by the number of times t_1 and t_2 co-occurred, i.e., $w(t_1, t_2) = |\{(u, s) \in U \times S | t_1, t_2 \in T_{us}\}|$. The relatedness between tags is defined directly by the edge weights. Given a tag $t \in T$, the tags that are most related to it are all the tags $t' \in T$ with $t \neq t'$

such that $w(t, t')$ is maximal. Table I illustrates five tags and their five most related ones; they were calculated based on the MovieLens dataset[10], which includes tags associated to a variety of movies.

Table I. Tags relatedness based on their co-occurrence.

Tag	1	2	3	4	5
superhero	comic book	marvel	super-hero	DC	Batman
opera	18th century	Mozart	Oscar (Best Actor)	backstage	music
holocaust	World War II	true story	nazi	Nazis	Poland
love	surreal	death	books	friendship	relationships
drugs	crime	violence	heroin	addiction	organized crime

We extend our tag cloud $cloud_t(u, r)$ by incorporating to it a folksonomy which is created based on the P most related tags associated to t_u; at the same time, we maintain its original rating r to the related tags. For instance, if user u has tagged a movie s with t_{us} = 'superhero', and rated s with $r = 3.5$, her augmented tag cloud $cloud^*_t(u, r)$ will contain the tags superhero, comic book, marvel, super-hero, DC and Batman associated to the same rating $r = 3.5$ (assuming $P = 5$).

5.5 Face Cloud

A face cloud corresponds to detected and recognized faces which are of interest to the user. By making a scribble around an actor's face, it is possible to infer that such person captured the user attention at that moment. On the client side, the recognition process can be accomplished automatically, when there are enough samples on the server to support the recognition; or manually, when the user herself provides the identification of that face.

We define $cloud_f(u, r)$ as the **rating face cloud** for a given user u and rating r, containing the set of pairs $(f_u, n_{f_u,u,r})$, where $f_u \in F$ is a face scribbled/recognized by user u and $n_{f_u,u,r} = |\{s \in S_u | f_u \in F_{us} \ \ \& \ \ \delta_u(s) = r\}|$. Thus, $n_{f_u,u,r}$ contains the frequency of occurrence of face f_u for all movies that user u has associated with rating r.

Analogously to tag cloud, we also provide an augmentation to the face cloud defined above. Another folksonomy is created, but in this case, each vertice of the graph is a recognized face by the user. If user u scribbled/recognized f_1 = 'Ben Stiller' and f_2 = 'Teri Polo' in the same movie s, so both actors will have and edge, and its weight $w(f_1, f_2)$ will be the number of times f_1 and f_2 co-occurred in S. As a result, the folksonomy will contain a list of persons and their relations, which can dictate how much two actors are linked to each other.

We create the augmented face cloud $cloud^*_f(u, r)$ in the same way it was done with tag cloud. The advantage of having this extension is that the profile of a user who recognized only a few actors will be as rich as the profile of other users who recognized more faces, maintaining, of course, each user's preference or level of interest defined by the assigned rating.

5.6 Users Similarity

The adopted similarity measure between two users follows the Pearson correlation coefficient, as firstly depicted by Resnick et al. [Resnick et al. 1994]. Let S_{uv} be the set of all items rated by both users u and v, i.e., $S_{uv} = \{s \in S | \delta_u(s) \neq \varnothing \ \ \& \ \ \delta_v(s) \neq \varnothing\}$. The similarity function $sim(u, v)$ is defined as:

$$sim(u, v) = \left(\frac{|S_{uv}|}{|S|} \right) \times \left(\frac{\sum_{s \in S_{uv}} (\delta_u(s) - \bar{\delta}_u)(\delta_v(s) - \bar{\delta}_v)}{\sqrt{\sum_{s \in S_{uv}} (\delta_u(s) - \bar{\delta}_u)^2 \sum_{s \in S_{uv}} (\delta_v(s) - \bar{\delta}_v)^2}} \right) , \quad (1)$$

[10] http://www.grouplens.org/node/12/

where $\bar{\delta}_u$ is the rating values mean for all items rated by user u, and $\bar{\delta}_v$ the same for user v. We adopted the similarity function defined in Equation 1 because it considers the fact that users may use the rating scale differently. Instead of using the absolute values of ratings, we consider their deviations from the average rating of the corresponding user [Adomavicius and Tuzhilin 2005]. In addition, the division $|S_{uv}|/|S|$ is used to give more confidence to those users who have more movies in common.

6. RECOMMENDATION

This section presents one of the services showed on the personalization module illustrated in Figure 1 (d), which is a recommender system that exploits metadata created by users. First, we describe in the next three subsections a set of recommendation algorithms that were previously reported in the literature; the last one, in turn, describes our proposed approach.

6.1 Content-Based Filtering

Szomszor et al. [Szomszor et al. 2007] proposed a content-based filtering approach which uses keywords to predict the ratings a user would give to new movies. They named this algorithm as **Weighted Keyword Cloud Comparison**[11] and its definition is based on a measure of keyword cloud similarity which considers weights at the movie's keywords level and at the keyword cloud level. Given an unrated movie s, they consider the set of keywords K_s and calculate its similarity to $cloud_k(u,r)$ as:

$$\sigma_k(u,s,r) = \sum_{\{(k,n_{k,u,r}) \in cloud_k(u,r) | k \in K_s\}} \frac{n_{k,u,r}}{\log(N_k)} \quad . \tag{2}$$

It means that they sum over all keywords that K_s and the keyword cloud $cloud_k(u,r)$ have in common, weighting each keyword k proportionally to its frequency $n_{k,u,r}$ in the keyword cloud, and inversely proportional to the logarithm of its global frequency N_k. The keyword-weighted average rating $\lambda_k(u,s)$ is then defined as:

$$\lambda_k(u,s) = \frac{\sum_{r \in R} \sigma_k(u,s,r) r}{\sum_{r \in R} \sigma_k(u,s,r)} \quad , \tag{3}$$

which is used in combination with the movie average rating $\bar{\delta}(s)$:

$$\bar{\delta}(s) = \frac{1}{U_s} \sum_{v \in U_s} \delta_v(s) \quad , \tag{4}$$

in order to predict the user rating for the movie s:

$$\delta_u(s) = (1-\gamma)\bar{\delta}(s) + \gamma \lambda_k(u,s) \quad , \tag{5}$$

where $0 < \gamma < 1$ is the factor weighting the contribution of the two estimates. Analogously to the authors, we set $\gamma = 1/2$, and the predicted rating r for movie s is the nearest value in R according to $\delta_u(s)$.

6.2 Collaborative-Based Filtering

Collaborative filtering algorithms make rating predictions based on the entire collection of previously rated items by the users. In other words, the value of an unknown rating $\delta_u(s)$ for user u and item s

[11] In fact, the original name proposed by the authors is **Weighted Tag Cloud Comparison**; but they use the same keywords gathered from the IMDB archive. As we are also considering the notion of tag cloud as tags created by the user, we preferred to rename the algorithm in order to make a distinction between the approaches.

is usually calculated as an aggregate of the ratings of the Q most similar users for the same item s. Let us consider as $\hat{U}$ the set of Q users that are the most similar to user u and who have rated item s. We define the aggregation function as [Adomavicius and Tuzhilin 2005]:

$$\delta_u(s) = \bar{\delta}_u + \frac{\sum_{v \in \hat{U}} sim(u,v) \times (\delta_v(s) - \bar{\delta}_v)}{\sum_{v \in \hat{U}} |sim(u,v)|} \quad . \tag{6}$$

Once again, we consider the rating values' deviations from the average rating of the corresponding user in order to address the issue of having different rating scales to each different user.

6.3 Hybrid-Based Filtering

A hybrid recommender system combines collaborative and content-based methods, trying to avoid limitations inherent to both approaches. Section 2 discussed some of these limitations, together with the description of a set of ways to combine them into a unique hybrid approach. In this article, for the sake of simplicity, we adopt a linear combination approach by extending the content-based algorithm presented in Subsection 6.1. Specifically, Equation 5 is slightly modified to substitute the movie average rating $\bar{\delta}(s)$ by the collaborative-based filtering prediction defined in Subsection 6.2. Thus:

$$\delta_u(s) = (1-\gamma) Collaborative_u(s) + \gamma \lambda_k(u,s) \quad , \tag{7}$$

where $Collaborative_u(s)$ is equivalent to Equation 6 and $\gamma = 1/2$.

6.4 Annotation-Based Filtering

This section presents the recommendation algorithm proposed in this article. Let us consider $match_t(u,s)$ as the set of pairs $(t, n_{t,u,r}) \in cloud_t^*(u,r)$ where $t \in T_s \cap T_u$; and $match_f(u,s)$ as the set of pairs $(f, n_{f,u,r}) \in cloud_f^*(u,r)$ where $f \in F_s \cap F_u$. Based on them, we define the tag and face-weighted average ratings as:

$$\lambda_t(u,s) = \sum_{match_t(u,s)} \frac{\sum_{r \in R} n_{t,u,r}\ r}{\sum_{r \in R} n_{t,u,r}} \quad , \quad \lambda_f(u,s) = \sum_{match_f(u,s)} \frac{\sum_{r \in R} n_{f,u,r}\ r}{\sum_{r \in R} n_{f,u,r}} \quad . \tag{8}$$

From these two average ratings, we combine them together to predict a rating value based solely on the user annotations:

$$\lambda_{tf}(u,s) = \frac{\lambda_t(u,s) + \lambda_f(u,s)}{|match_t(u,s)| + |match_f(u,s)|} \quad . \tag{9}$$

One problem of using only $\lambda_{tf}(u,s)$ is that some users may not provide any annotation. In this case, the divisor of the above division will be zero. In order to avoid this problem, we combine $\lambda_{tf}(u,s)$ with the content-based filtering algorithm defined previously, i.e.:

$$\delta_u(s) = (1-\gamma) Content_u(s) + \gamma \lambda_{tf}(u,s) \quad , \tag{10}$$

where $Content_u(s)$ is equivalent to Equation 5, $\gamma = 0$ if $|match_t(u,s)| + |match_f(u,s)| = 0$ and $\gamma = 1/2$ otherwise.

We will show in Section 7 that Equation 10 can already improve the results obtained by using only the content-based filtering. Before that, however, we follow to present an enhanced hybrid approach, which combines the above algorithm with the collaborative-based filtering. It is defined as:

$$\delta_u(s) = \begin{cases} Collaborative_u(s) & \text{if } |K_s| < \alpha \text{ and } |T_s| < \alpha \\ Annotation_u(s) & \text{if } \delta_v(s) = \varnothing \text{ or } |S_{uv}|/|S| < \beta \\ (1-\gamma)Collaborative_u(s) + \gamma Annotation_u(s) & \text{otherwise,} \end{cases} \quad (11)$$

where $Annotation_u(s)$ is equivalent to Equation 10, α and β are thresholds, and $\gamma = 1/2$. By analyzing the amount of metadata for each movie s, and the similarity parameters between users u and $v \in \hat{U}$, we choose among each of the approaches to make the prediction for s.

We can go a little further and explore the aforementioned algorithms with the genres metadata available for each movie. In doing so, we are able to provide better recommendations, as presented in Section 7. This enhancement is accomplished by considering $match_g(u, s)$ as the set of pairs $(g, n_{g,u,r}) \in cloud_g(u, r)$, where $g \in G_s \cap G_u$. Based on this set, we define the genre-weighted average rating as:

$$\lambda_g(u, s) = \sum_{match_g(u,s)} \frac{\sum_{r \in R} n_{g,u,r}\ r}{\sum_{r \in R} n_{g,u,r}} \quad , \quad (12)$$

and from this average rating, we combine it with the enhanced hybrid-based approach, as follows:

$$\delta_u(s) = (1-\gamma)Hybrid_u(s) + \gamma\lambda_g(u, s) \quad , \quad (13)$$

where $Hybrid_u(s)$ is equivalent to Equation 11, and $\gamma = 1/2$.

7. EXPERIMENTAL RESULTS

This section provides the results of the annotation-based filtering approach proposed in this article. It consists of comparing the performance of all recommender algorithms described in last section, using the root mean squared error (RMSE) metric [Anderson and Woessner 1992]. It indicates that the smaller the RMSE, the more accurate is a set of predictions. Given a set of N predicted ratings $\{r_i\}$ and the corresponding set of actual ratings $\{r_i^*\}$, the RMSE is defined as:

$$\text{RMSE}(\{r_i\}, \{r_i^*\}) = \sqrt{\frac{1}{N}\sum_i (r_i - r_i^*)^2} \quad (14)$$

The samples that were used as input data for evaluation correspond to a subset of the MovieLens dataset[12], which includes movies ratings of several real users, along with tags assigned by those users to a variety of movies. In the original dataset, there are about 70.000 users, who assign tags and rates to a set of 65.133 different movies, totalizing 95.580 tags and 10.000.054 ratings. However, as calculating predictions to all these users would be a very time-consuming task, we have randomly selected 500 users to be used in our evaluation, corresponding to 197.211 ratings.

We decided to adopt the MovieLens dataset instead of evaluating the performance with the use of the M4Note tool because the main objective of this study is to show how better the proposed recommender algorithm can perform with a large scale set of users. Thus, albeit we are dealing with real users, in fact, they did not use the M4Note tool to watch, annotate and assign rates for each movie. On the other hand, all annotations that could be created using the M4Note tool are simulated as described in the following paragraphs.

Hierarchical descriptions of each movie were explored in this work as explained in Subsection 4.1. We chose to use those from the IMDB archive, so each of the 65.133 movies from the MovieLens

[12]http://www.grouplens.org/node/12/

dataset was linked to the corresponding one in the IMDB dataset[13]; in this way, the lists of genres and keywords for each movie were used in our evaluation.

In addition to genres and keywords, which are appraised as hierarchical metadata in this evaluation, we also consider peer-level annotations produced by all 500 users existent in our dataset. Assigned tags and scribbled/recognized faces of interest are explored; but in the last case, as we do not have this information in the original datasets, we have simulated the user action of scribbling/recognizing a face by using her tags content to search for actors/actresses' names into the IMDB archive. Thus, when a tag is found in the IMDB's list of actors/actresses, we conceive it as a face recognized by the user. As a total, 12.758 faces were considered in the evaluation.

The set of 197.211 ratings was split into training and test sets. Thus, we assumed our training set corresponds to items watched so far by the users, being possible to create profiles of interest for each one, and our test set, in turn, is used to test our predictions against its actual ratings. This division was made in a way to have exactly 10 ratings for each user in the test set. Consequently, the training and test sets contain 192.211 and 5.000 ratings, respectively.

Besides the parameters values already defined so far, in this evaluation we have configured the others as follows: during tag and face clouds augmentation, $P = 5$; in the collaborative filtering, $Q = |\hat{U}| = 10$; and in the enhanced hybrid-based filtering, $\alpha = 5$ and $\beta = 0.02$. All these values were calculated experimentally, i.e., we analyzed different values for each parameter, and created a relationship among the obtained results with the amount of data in the data clouds, the viability of content metadata and the number of movies in common.

We ran all recommendation algorithms described in Section 6 with the same dataset. Figure 4 presents the obtained results. Graphs from (a) to (g) show the RMSE for all users, along with their number of ratings. For those algorithms that deal with collaborative filtering (graphs (d)-(g)), it is possible to visualize a tendency for better performance as more ratings are assigned by each user. This happens because the relatedness among users gets stronger as more ratings are available to be used by the similarity metric.

When evaluating the results for isolated users in the content-based and collaborative-based approaches (graphs (b) and (d)), one can note cases where overspecialization and new user problem are evident. The user who rated around 2250 movies obtained a bad score of about 2.4 of RMSE with the content-based, what suggests her profile contains lots of information (keywords/ratings), but not implying good predictions. Although this same user obtained around 1.4 of RMSE with the collaborative-based, in this same approach there are other users who rated a few movies (around 10), and obtained very poor predictions (about 3.8 of RMSE in the worst case), suggesting the occurrence of the new user problem.

The hybrid approach (graph (e)) was able to balance such content-based and collaborative-based results. Nevertheless, the same new users achieved around 2.4 of RMSE, which is worse than the enhanced hybrid approach, whose same new users obtained around 2.1 of RMSE in the worst case. This can be explained by the user profiling augmentation, which provides semantically richer concepts to the tag/face clouds, even with a few number of annotations and ratings produced by these users.

Figure 4 (h) presents the average RMSE for each approach. One can note that the annotation-based achieved better performance than content-based, improving its RMSE in about 3.9%. The enhanced hybrid-based, which considers users annotations, was also able to improve the traditional hybrid approach in about 2.7%. The best algorithm, however, was a combination of the enhanced hybrid-based with genres metadata available for each movie, achieving a score of 0.8886 of RMSE. Such approach improved the results of enhanced hybrid-based in about 3.3%.

From this analysis, we clearly see that the more information is available about users and content,

[13] http://www.imdb.com/interfaces/

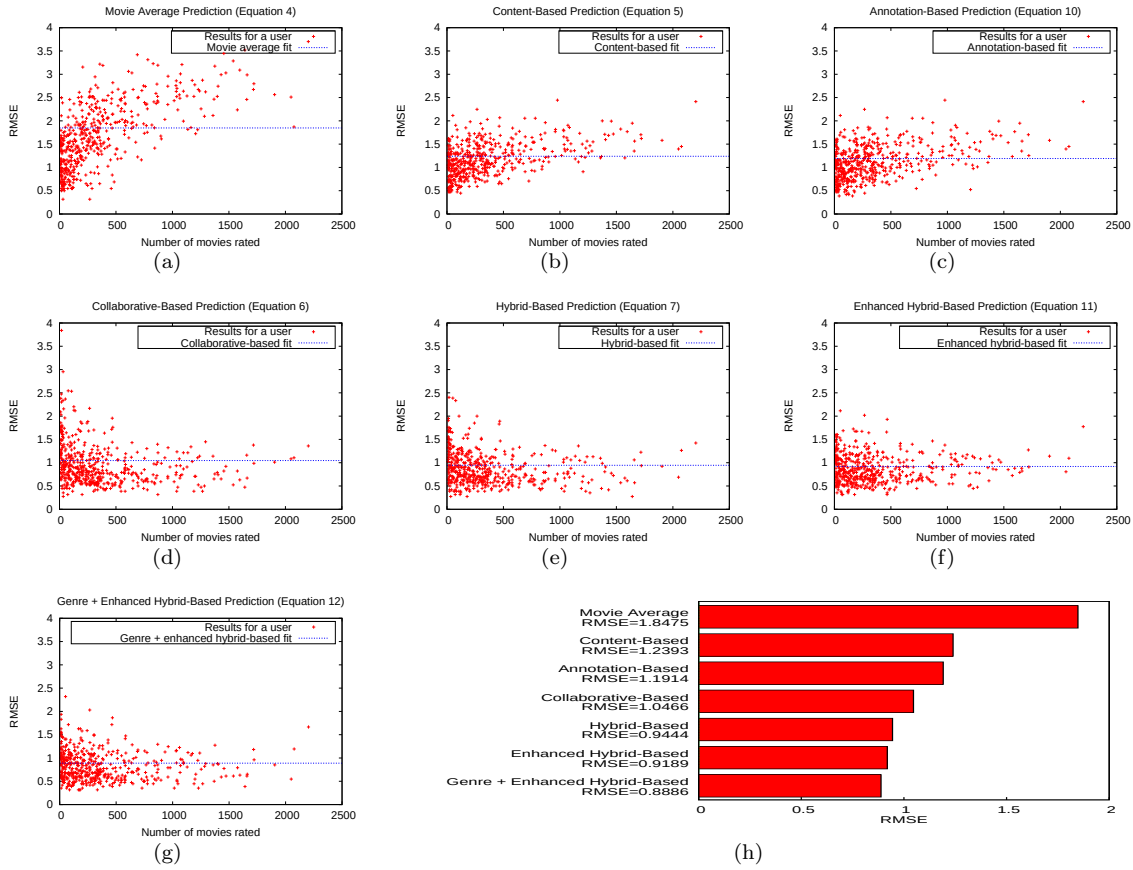

Fig. 4. Results for all considered recommendation algorithms. Graphs from (a) to (g) show the RMSE for all considered users along with their number of ratings: (a) is the movie average prediction (Equation 4), (b) is the content-based prediction (Equation 5), (c) is the annotation-based prediction (Equation 10), (d) is the collaborative-based prediction (Equation 6), (e) is the hybrid-based prediction (Equation 7), (f) is the enhanced hybrid-based prediction (Equation 11) and (g) is the genre plus enhanced hybrid-based prediction (Equation 12). Graph (h) shows the RMSE average for each approach.

the better the prediction algorithms will perform. We note that annotations are important to gather user's preferences and interests, but they depend on the available hierarchical content metadata in order to be helpful. This affirmative can be supported by the fact that not all users will provide tags and/or faces of interest, being necessary, in this case, the use of other methodologies, such as collaborative filtering or content-based algorithms. On the other hand, when those annotations are available, it is possible to combine them with relatedness measures in order to enrich the data clouds, and consequently, provide better recommendations with higher semantic level.

8. FINAL REMARKS

This article presented the Peersommender architecture, which considers user annotations in order to provide recommendations according to user's preferences. We have presented all its involved modules, which are essential steps to achieve good personalization: i) the hierarchical content description, using metadata available on centralized databases, such as IMDB; ii) the viability of enrichment tools to provide users with annotation mechanisms; iii) the profiling procedure, which constructs data clouds containing rich semantic information about user's preferences; and iv) the proposed recommender algorithm, which combines hierarchical and peer-level annotations, providing better predictions as

showed on the experimental evaluation described in this article.

The evaluation was executed over a large dataset containing real users, and it has demonstrated that better results can be achieved when combining impartial and organized metadata (such as keywords and associated genres) with information provided by users (such as ratings, tags and faces of interest). Thus, the contributions of this article can be summarized as follows:

—The user profiling mechanism, which considers the notion of data clouds containing different pieces of information about user's preferences. This profiling procedure, indeed, is augmented with statistical methods based on co-occurrence of tags and faces, resulting in a semantically richer profile, containing more data about the interests of the user.

—The annotation-based filtering approach, and its combination with collaborative-based, providing an enhanced hybrid approach that performs better when compared to previously proposed algorithms.

As future work, we plan to improve our recommender system by analyzing other types of annotation provided by the user, such as recognized speech, object recognition and image processing of captured frames. All this information shall be included in the user profile, in order to make possible the knowledge of more concepts of interest to the user. In addition, we plan to analyze how peer-level annotations can be used to help content providers during the creation of metadata (Figure 1 (e)). This process must be able to decide whether a user annotation may be considered impartial data, so that it can describe the content without denoting user's thoughts.

ACKNOWLEDGMENTS

We would like to thank the valuable contributions from the CWI team, in particular Dick Bulterman, Pablo Cesar and Jack Jansen.

REFERENCES

Adomavicius, G. and Tuzhilin, A. 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): 734–749, 2005.

Anderson, M. P. and Woessner, W. W. *Applied Groundwater Modeling: Simulation of Flow and Advective Transport*. Academic Press, 1992.

Angeletou, S., Sabou, M., and Motta, E. Folksonomy Enrichment and Search. In *6th. European Semantic Web Conference*, L. Aroyo, P. Traverso, F. Ciravegna, P. Cimiano, T. Heath, E. Hyvonen, R. Mizoguchi, E. Oren, M. Sabou, and E. P. B. Simperl (Eds.). Lecture Notes in Computer Science, vol. 5554. Springer, pp. 801–805, 2009.

Balabanovic, M. and Shoham, Y. Fab: Content-Based, Collaborative Recommendation. *Communications of the ACM* 40 (3): 66–72, 1997.

Bell, R. M. and Koren, Y. Lessons from the Netflix Prize Challenge. *ACM SIGKDD Explorations Newsletter* 9 (2): 75–79, 2007.

Breese, J. S., Heckerman, D., and Kadie, C. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In *Proceedings of the 14th. Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, Madison, WI, USA, pp. 43–52, 1998.

Bulterman, D. C. A. Animating Peer-Level Annotations Within Web-Based Multimedia. In *7th Eurographics Workshop on Multimedia*. Nanjing, China, pp. 49–57, 2004.

Cattuto, C., Benz, D., Hotho, A., and Stumme, G. Semantic Grounding of Tag Relatedness in Social Bookmarking Systems. In *The Semantic Web – ISWC 2008*, A. P. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. W. Finin, and K. Thirunarayan (Eds.). Lecture Notes in Computer Science, vol. 5318. Springer, Berlin/Heidelberg, pp. 615–631, 2008.

Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D., and Sartin, M. Combining Content-Based and Collaborative Filters in an Online Newspaper. In *Proceedings of ACM SIGIR Workshop on Recommender Systems*. Berkeley, California USA, pp. 1–11, 1999.

Cohen, W. W., Schapire, R. E., and Singer, Y. Learning to Order Things. *Journal of Artificial Intelligence Research* vol. 10, pp. 243–270, 1999.

Gauch, S., Speretta, M., Chandramouli, A., and Micarelli, A. User Profiles for Personalized Information Access. In *The Adaptive Web*, P. Brusilovsky, A. Kobsa, and W. Nejdl (Eds.). Lecture Notes in Computer Science, vol. 4321. pp. 54–89, 2007.

Gemmis, M. D., Lops, P., Semeraro, G., and Basile, P. Integrating Tags in a Semantic Content-based Recommender. In *Proceedings of the 2008 ACM Conference on Recommender Systems*. Lausanne, Switzerland, pp. 163–170, 2008.

Goularte, R., Camacho-Guerrero, J. A., Inacio Jr., V. R., Cattelan, R. G., and Pimentel, M. G. C. M4Note: a Multimodal Tool for Multimedia Annotations. In *Proceedings of 10th Brazilian Symposium on Multimedia and the Web & 2nd Latin American Web Congress*. Ribeirao Preto, SP, Brazil, pp. 142–149, 2004.

Halpin, H., Robu, V., and Shepherd, H. The Complex Dynamics of Collaborative Tagging. In *Proceedings of the 16th International Conference on World Wide Web*. Banff, Alberta, Canada, pp. 211–220, 2007.

Heckner, M., Neubauer, T., and Wolff, C. Tree, funny, to_read, google: What are Tags Supposed to Achieve? A Comparative Analysis of User Keywords for Different Digital Resource Types. In *Proceeding of the 2008 ACM Workshop on Search in Social Media*. Napa Valley, California, USA, pp. 3–10, 2008.

Koren, Y., Bell, R., and Volinsky, C. Matrix Factorization Techniques for Recommender Systems. *IEEE Computer* 42 (8): 30–37, 2009.

Manzato, M. G., Coimbra, D. B., and Goularte, R. Multimedia Content Personalization Based on Peer-level Annotation. In *Proceedings of the 7th. European Conference on Interactive TV*. Leuven, Belgium, pp. 57–66, 2009.

Manzato, M. G. and Goularte, R. Supporting Multimedia Recommender Systems with Peer-level Annotations. In *Proceedings of the 15th. Brazilian Symposium on Multimedia and the Web*. Fortaleza, CE, Brazil, pp. 202–209, 2009.

Patel, S. N. and Abowd, G. D. The ContextCam: Automated Point of Capture Video Annotation. In *UbiComp 2004: Ubiquitous Computing*, F. Khendek and R. Dssouli (Eds.). Lecture Notes in Computer Science, vol. 3205. Springer, pp. 301–318, 2004.

Pazzani, M. A Framework for Collaborative, Content-Based, and Demographic Filtering. *Artificial Inteligence Review* 13 (5-6): 393–408, 1999.

Pazzani, M. and Billsus, D. Learning and Revising User Profiles: The Identification of Interesting Web Sites. *Machine Learning* 27 (3): 313–331, 1997.

Popescul, A., Popescul, R., Ungar, L. H., Pennock, D. M., and Lawrence, S. Probabilistic Models for Unified Collaborative and Content-Based Recommendation in Sparse-Data Environments. In *Proceedings of the 17th. Conference on Uncertainty in Artificial Intelligence*. Seattle, Washington, USA, pp. 437–444, 2001.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*. Chapel Hill, North Carolina, United States, pp. 175–186, 1994.

Schein, A. I., Popescul, A., Ungar, L. H., and Pennock, D. M. Methods and Metrics for Cold-Start Recommendations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM Press, Tampere, Finland, pp. 253–260, 2002.

Sen, S., Vig, J., and Riedl, J. Tagommenders: Connecting Users to Items through Tags. In *Proceedings of the 18th International Conference on World Wide Web*. Madrid, Spain, pp. 671–680, 2009.

Soboroff, I. and Nicholas, C. Combining Content and Collaboration in Text Filtering. In *Proceedings of the IJCAI-99 Workshop on Machine Learning for Information Filtering*. Stockholm, Sweden, pp. 86–91, 1999.

Szomszor, M., Cattuto, C., Alani, H., O'Hara, K., Baldassarri, A., Loreto, V., and Servedio, V. D. Folksonomies, the Semantic Web, and Movie Recommendation. In *4th European Semantic Web Conference, Bridging the Gap between Semantic Web and Web 2.0*. Innsbruck, Austria, pp. 71–84, 2007.

Tso-Sutter, K. H. L., Marinho, L. B., and Schmidt-Thieme, L. Tag-aware Recommender Systems by Fusion of Collaborative Filtering Algorithms. In *Proceedings of the 2008 ACM Symposium on Applied Computing*. Fortaleza, CE, Brazil, pp. 1995–1999, 2008.

Venkatesh, S., Adams, B., Phung, D., Dorai, C., Farrell, R. G., Agnihotri, L., and Dimitrova, N. "You Tube and I Find" – Personalizing Multimedia Content Access. *Proceedings of the IEEE* 96 (4): 697–711, 2008.

Zanardi, V. and Capra, L. Social Ranking: Uncovering Relevant Content Using Tag-based Recommender Systems. In *Proceedings of the 2008 ACM Conference on Recommender Systems*. Lausanne, Switzerland, pp. 51–58, 2008.