Hybrid Content and Tag-based Profiles for Recommendation in Collaborative Tagging Systems

Daniela Godoy and Analía Amandi

ISISTAN Research Institute, UNICEN University
Campus Universitario, Paraje Arroyo Seco
CP 7000, Tandil, Bs. As., Argentina
Also at CONICET, Consejo Nacional de Investigaciones Científicas y Técnicas
{dgodoy,amandi}@exa.unicen.edu.ar

Abstract

Collaborative tagging systems have grown in popularity over the Web in the last years on account of their simplicity to categorize and retrieve content using open-ended tags. The increasing number of users providing information about themselves through social tagging activities caused the emergence of tag-based profiling approaches, which assume that users expose their preferences for certain contents through tag assignments. On the other hand, numerous content-based profiling techniques have been developed to address the problem of obtaining accurate models of user information preferences in order to assist users with information-related tasks such as Web browsing or searching. In this paper we propose a hybrid user profiling strategy that takes advantage of both content-based profiles describing long-term information interests that a recommender system can acquired along time and interests revealed through tagging activities, with the goal of enhancing the interaction of users with a collaborative tagging system. Experimental results of using hybrid profiles for tag recommendation are reported and possible applications of these profiles for obtaining personalized recommendations in collaborative tagging systems are discussed.

1. Introduction

User profiling is an essential component of personal information agents and recommendation systems in general. Content-based recommendation approaches rely on profiles containing accurate representations of user interests, preferences and habits used to satisfy long-term information needs. The problem of learning and adapting user profiles starting from the observation of the browsing history or documents read by the user has been addressed in several works [23, 7, 33].

Recently, collaborative or social tagging sites such as del.icio.us\(^1\), Technorati\(^2\) or Flickr\(^3\) have achieved widespread success on the Web due to their simplicity to browse and search an important body of shared resources. In these sites, users annotate resources such as Web pages, blog posts or pictures using a freely chosen set of keywords or tags. In contrast to the use of pre-defined taxonomies to categorize content, a social classification scheme commonly known as folksonomy [19] is expected to emerge from the collective behavior of users.

The activities carried out by users in social tagging systems, including posting resources or assigning tags to resources, have become a novel source of information about user interests. In this regard, tag-based profiles attempt to capture interests based on the observation of tag assignments in folksonomies. Most of these profiles use a single group of popular tags or tag-cloud to represent the user interests [25, 4, 29]. However, a unique tag-cloud results insufficient to represent the diverse interests users might have across several domains. Other approaches [32, 30] have considered multiple interests by mining folksonomies for extracting topics a user is interested in.

In this paper we propose to integrate content-based profiles representing long-term user interests gathered by recommenders through observation of browsing activities with tag-based profiles acquired by capturing the user interaction with one or more collaborative tagging systems. Thus, tag-based profiles can be enriched with the knowledge of information preferences recommenders or personal agents assisting a user might have acquired over time without the

\(^1\)http://del.icio.us/
\(^2\)http://technorati.com/
\(^3\)http://www.flickr.com/
need of running an intensive knowledge discovery process over folksonomies. Hybrid profiles resulting of this integration can be exploited to assist users in finding resources, people or tags within social tagging systems.

The paper is organized as follows. Section 2 discusses current works in tag-based profiling. Section 3 presents the proposed approach to relate tagging activities to pre-existent representations of long-term user preferences in order to create hybrid user profiles. Experimental results of exploiting these profiles for obtaining personalized tag recommendations are reported in Section 4. Finally, concluding remarks are stated in Section 5.

2. Related Works

Most methods for building tag-based profiles extract tags users associate to resources in folksonomies considering more frequently chosen tags as more important for describing user interests. In [25], a vector of weighted tags is obtained using tag frequency of occurrence in the resources a user tagged and it is applied to rank Web search results according to their similarity with this tag vector. TBPProfile [4] also uses weighted vector of tags to represent user interests, but tag weights are based on inverse user frequency. This system locales like-minded users by comparing their corresponding vectors using the standard cosine similarity measure. In a similar way, tags weighted according to the fraction they covered of the user tagging actions are used in [29] to generate hot-lists of recommendations by combining tags and item overlapping for constructing per-tag common interest networks.

Firan et al. [6] investigated tag-based user profiles in contrast to conventional profiles based on song and track usage in the music search portal Last.fm⁴. The tagging frequency on a track owned by a user is used in this work to determine relevant tags and their associated scores, then a search-based method employs tags to find similar users and recommend interesting songs. Compared with conventional track-based recommender approaches, tag-based profiles significantly improve the quality of recommendations.

Using a single vector of weighted tags to represent the user interests has some drawbacks. First, more frequent tags tend to be too general for describing interests so that profiles tend to lose specificity. More importantly, users usually have diverse interests spanning across different domains that can not be embraced by a unique vector or tag-cloud. To overcome this problem Au Yeoung at al. [32] proposed an algorithm which performs graph-based clustering over the network of user tagged documents to identify interest topics and extract tag vectors for them. Multiple tag-clouds were also considered in [30], which enriches descriptions of movie titles with user interests and opinions taken from a folksonomy for predicting the rating of an unseen movie. For each rating belonging to a predefined scale, a tag-cloud is assigned to the rating with the keywords belonging to all movies that the user has associated with that rating so that predictions for unrated movies are made based on their similarity to rating tag-clouds.

Michlmayr et al. [21] compared the method for tag-based profile construction based on a single vector of weighted tags, which the authors called the naive approach, with two different approaches, one based on co-occurrence and one based on adaptation. In the co-occurrence approach, tags frequently used in combination for annotating objects are assumed to have certain semantic relation. Thus, profiles are represented by graphs in which nodes correspond to tags and edges denote relationships among them. The adaptive approach, implemented in Add-A-Tag [22] algorithm, extends this model to include temporal information by updating the weights of edges in the graph using an evaporation technique known from ant algorithms for discrete optimization. Tag-based profiles built using both co-occurrence and adaptive approaches had better acceptance from users than those built with the naive approach. The idea of using semantic relationships among tags in tag-based profiles has also been explored in [10], in which the semantic distance between two tags is calculated based on co-occurrence statistics and common sense reasoning using ConceptNet [17].

In other line of research, several works have proposed to mine folksonomies to glean collective knowledge potentially useful for recommendation. Li et al. [13], for instance, propose a tag-based approach for discovering social interests shared by groups of users. First, association rule mining algorithms are used to discover patterns in tag assignments that can be used to characterize interest topics and, then, users and URLs are clustered according to these topics. If a set of tags is frequently used by a group of users, these users are supposed to form an interest community. Tag data and image visual features are used in [11] to extract representative images for landmarks in a combined context and content-based analysis of community-contributed datasets such as Flickr pictures. FolkRank [9] tries to rank users, tags and resources according to their importance in folksonomies implementing a weight-spreading ranking scheme. These approaches rely on gathering collective knowledge rather than personalized user preferences.

User modeling is an activity that can clearly benefit from inferring knowledge about users based on their tagging activities. Carmagnola et al. [2] take into account several tag properties to define tag categories and provide some insights on how these categories can be related to user modeling dimensions, including interactivity, organization and content identification. However, the free-form nature of tagging also leads to a number of problems, such as ambiguity.

⁴http://www.last.fm/
and synonymy among others, which can be alleviated by contextualizing tags based on the knowledge of user information preferences.

Several user profiling techniques have been developed to address the problem of obtaining accurate models of user information preferences to help users with information-related tasks [7], yet only a few efforts have been made on extending available content-based profiling approaches in order to exploit tagging activity. In this direction, a method for integrating tagging history from several Web 2.0 sites into their ontological profiles by matching tags with ontology concepts has been proposed in [1]. In this work, three steps have to be carried out to map tags to ontological concepts: tag filtering, acquisition of semantic information about tags to establish a common vocabulary, and categorization of the obtained concepts according to existing ontology classes. This method showed promising results in News@hand, a news recommender system.

In contrast to the mentioned approaches that involve mining folksonomies for detecting topics that can be related with user interests or exploiting linguistic knowledge to relate tags with concepts in an ontology, we propose to combine content-based profiles gathered by personal agents, also known as interface agents [18, 15], with information about tagging activities expressed by users in social tagging systems.

3. Hybrid User Profiles

Folksonomies are the primary structure underlying collaborative tagging systems. Formally, a folksonomy can be defined as a tuple \( F := (U, T, R, Y, \prec) \) which describes the users \( U \), resources \( R \), and tags \( T \), and the user-based assignment of tags to resources by a ternary relation between them, i.e. \( Y \subseteq U \times T \times R \) [9]. In this folksonomy, \( \prec \) is a user-specific sub-tag/super-tag-relation possibly existing between tags, i.e. \( \prec \subseteq U \times T \times T \).

The collection of all tag assignments of a single user constitutes a persononomy, i.e. the persononomy \( \mathbb{P}_u \) of a given user \( u \in U \) is the restriction of \( F \) to \( u \), i.e., \( \mathbb{P}_u := (T_u, R_u, I_u, \prec_u) \) with \( I_u := \{(t, r) \in T \times R | (u, t, r) \in Y \} \), \( T_u := \pi_1(I_u) \), \( R_u := \pi_2(I_u) \), and \( \prec_u := \{(t_1, t_2) \in T \times T | (u, t_1, t_2) \in \prec \} \), where \( \pi_1 \) denotes the projection on the 1th dimension. In social tagging systems, tags are used to organize shared information within a personal information space. Thus, other users can access a user persononomy by browsing and searching the entire folksonomy using the available tags.

Numerous personal agents and recommender systems have been developed to assist users with tasks such as searching and browsing the Web [16, 26]. In this paper we propose to take advantage of the knowledge gained by these agents through observation of user browsing behavior over long periods to generate personalized recommendations about users, resources or tags in folksonomies. For example, if a recommender detects that a page which is about to be tagged belongs to a certain interest category existing in the user profile, it can anticipate the tags the user is likely to assign to this page. Likewise, if a user is subscribed to a RSS feed for a given tag the interests described in the content-based profile can be used to contextualize and filter the incoming bookmarks.

In this work we considered content-based profiles providing a hierarchical representation of long-term user interests supporting the categorization of Web pages the user browsed through into semantically meaningful concepts. Ontology-based user profiling approaches [23, 14] as well as techniques based on clustering algorithms [12, 24] fall into this category. The user profiling approach used in this work to acquire content-based profiles based on clustering of the documents read by a user during browsing is described in the next section.

3.1. Content-Based User Profiles

User profiling is built upon a clustering algorithm, named WebDCC (Web Document Conceptual Clustering) [8], with structures and procedures specifically designed for user profiling. Modeling user interests by conceptual clustering allows agents to acquire descriptions of user interests without user intervention through non-intrusive observation of user activities. In the observation process, agents capture experiences regarding user interests such as Web pages a user read or bookmarked for future reading, read news, etc.

WebDCC carries out incremental, unsupervised concept learning over the collected experiences. First introduced in [20], conceptual clustering includes not only clustering, but also characterization, i.e. the formation of intentional concept descriptions for extensionally defined clusters. More formally, conceptual clustering is defined as the task of, given a sequential presentation of experiences and their associated descriptions, finding clusters that group these experiences into concepts or categories, a summary description of each concept and a hierarchical organization of them [31].

The advantage of using an algorithm belonging to the conceptual clustering paradigm is twofold. First, it is an incremental approach which allows agents interacting with users over time to acquire and maintain interest hierarchies as well as deal with subject areas that can not be predicted beforehand. Second, unlike most user profiling approaches, this algorithm offers a readable description of the user interests as a means of understanding the user information needs and explaining agent recommendations.

Experiences representing user interests that agents cap-
ture through observation, which are vector representations of information items (e.g., Web pages) based on the vector space model [28], are presented to WebDCC algorithm in an on-line fashion. Experiences are analyzed to learn a conceptual description of user interests and organized within user profiles to be applied in recommendation. Identification of categories or topics a user is interested in is based on clustering of similar past experiences. Thus, user profiles are built starting from scratch and constantly refined as new experiences representing user interests become available.

In order to cope with drifting interests, experiences have an associated relevance value that is used for both gaining confidence in those experiences that had led to successful recommendations and gradually losing confidence in irrelevant experiences that did not provide good recommendations, adapting the profile to recognize and eventually remove no longer interesting topics.

Hierarchies of concepts produced by this algorithm are classification trees in which internal nodes represent concepts and leaf nodes represent clusters of experiences. The root of the hierarchy corresponds to the most general concept, which comprises all the experiences the algorithm has seen, whereas inner concepts become increasingly specific as they are placed lower in the hierarchy, covering only subsets of experiences by themselves. Finally, terminal concepts are those with no child concepts but clusters.

More formally, a hierarchy consists of a number of concepts, denoted by $C = \{c_1, c_2, \ldots, c_n\}$, which are gradually discovered by the algorithm as new experiences become available. In order to automatically assign experiences to concepts, the algorithm associates each of them with a description given by a set of terms, $c_i = \langle\{t_1, w_1\}, \ldots, \{t_m, w_m\}\rangle$, weighted according to their importance in the concept summarization. This description constitutes a linear classifier for the category and emerges from observing the common features of experiences in the category and those a novel experience should have in order to belong to it.

WebDCC builds a hierarchical set of classifiers, each based on its own set of relevant features, as a combined result of a feature selection algorithm for deciding on the appropriate set of terms at each node in the tree and a supervised learning algorithm for constructing a classifier for such node. An instantiation of Rocchio classifier [27] in which the prototype of a category is the plain average of all training experiences, as it is defined in Equation 1, is used for learning classifiers.

Leaves in the hierarchy correspond to clusters of experiences belonging to all ancestor concepts. This is, clusters group highly similar experiences observed by the algorithm so that a set of $n_i$ experiences or documents belonging to a concept $c_i$ and denoted by $E_i = \{e_1, e_2, \ldots, e_{n_i}\}$ is organized into a collection of $k$ clusters, $U_{ji} = \{u_{1i}, u_{2i}, \ldots, u_{ki}\}$, containing elements of $E_i$ such that $u_{li} \cap u_{pi} = \emptyset, \forall l \neq p$.

WebDCC integrates classification and learning by sorting each experience through the concept hierarchy and simultaneously updating it. Upon encountering a new experience, the algorithm incorporates it below the root of the existing hierarchy and then recursively compares the experience to each child concept as it descends the tree. When the experience can not be further classified down, the algorithm considers whether to incorporate the experience into a cluster or create a new singleton cluster or category.

The incorporation of experiences to the hierarchy is followed by an evaluation of the hierarchical structure in order to determine whether new concepts can be created or some restructuring is needed. The formation of concepts is driven by the notion of conceptual cohesiveness. Highly cohesive clusters are assumed to contain similar experiences belonging to the same category, whereas clusters exhibiting low cohesiveness are assumed to contain experiences concerning distinctive aspects of more general categories. In the last case, a concept summarizing the cluster is extracted, enabling a new partitioning of experiences and the identification of sub-categories.

Finally, re-structuring operators such as merge, split and promote concepts are applied to reduce the effect of example ordering during learning. The merge operation takes two concepts and combines them into a single one, whereas splitting takes place when a concept is no longer useful to describe experiences in a category and then it can be removed. The promotion of concepts to become siblings of their parent concepts is also taken into account in order to place concepts at the appropriate hierarchical level.

### 3.2. Linking Tags to User Interests

In order to build hybrid profiles, categories representing user interests in content-based profiles are populated with the tags users frequently associate to resources in that categories. To accomplish this goal, tagged resources have to be first categorized according to the current representation of user interests given by the interest hierarchy.

Classification proceeds in a top-down manner based on the hierarchical organization of classifiers. Initially, a resource belonging to the root category is categorized by classifiers at the first level of the tree. Then, classifiers at the second level take the resource that has been classified in one of the previous level categories and classify it into categories at the second one. This procedure continues until the resource has reached some leaf node in the hierarchy or it cannot be further classified down.

Each node in the hierarchy acts as a linear classifier which is compared with the resource to be classified (bookmark or Web page), i.e. the prototype $p_{c_i}$ of a category $c_i$ is
a vector of weighted terms as follows:

\[ p_{ci} = \frac{1}{|c_i|} \sum_{d \in c_i} d \]  

(1)

where \( d \) are the documents belonging to the category \( c_i \) that were observed during learning by WebDCC algorithm or previously tagged resources. Thus, a resource is classified in a certain category if it exceeds a minimum similarity to the category prototype.

Once a resource has been classified into a given category in the hierarchy, the algorithm performs a centroid-based clustering to place it in the most appropriate cluster below this category. In order to predict which this cluster is, the closest centroid is determined by comparing the content of the resource with all centroids in the existing clusters. Given the cluster \( s_{ji} \) belonging to the category \( c_i \), which is composed of the vector representations corresponding to a set of documents, the centroid vector \( p_{s_{ji}} \) is defined as follows:

\[ p_{s_{ji}} = \frac{1}{|s_{ji}|} \sum_{d \in s_{ji}} d \]  

(2)

The distance measure determines the degree of resemblance between the vector representations of the resource and the cluster centroids and is calculated using the cosine similarity. As the result of resource comparison with the prototypes, the resource is assigned to the cluster with the closest centroid below the category \( c_i \), i.e.

\[ \arg \max_{j=1...k} \text{sim} \left( r, p_{s_{ji}} \right) \]  

(3)

provided that the similarity is higher than a minimum similarity threshold \( \delta \). Experiences no similar enough to any existent centroid according to this threshold cause the creation of new singleton clusters. The similarity threshold has a critical effect on the creation of the hierarchy, a value of \( \delta \) near to 0 will produce very few clusters (only one cluster when \( \delta = 0 \) grouping even highly dissimilar experiences or resources; whereas a value of \( \delta \) near to 1 will produce a large number of clusters (equal number of clusters that instances when \( \delta = 1 \)). In the experimental evaluation of the approach we studied the impact of variations in the value of this parameter on recommendation.

Until this point the procedure for adding a tagged resource to the hierarchy is the same as the one used to incorporate regular experiences in the profile (e.g. Web pages the user browsed through). For considering the tags the user assigned to the resource recently classified in addition to its content, tags associated to resources in a cluster need to be aggregated in the description of this cluster leading to a hybrid user profile.

For each cluster in the hierarchy, a set of the most frequently used tags is extracted to represent the corresponding tag assignment preferences for the experiences or resources belonging to this cluster. The set of tags related to a cluster \( s_{ji} \) within the category \( c_i \) can be defined within the personomies \( P_u \) as follows:

\[ T_{s_{ji}} = \{ t \in T \mid (t, r) \in I_u \land r \in s_{ji} \} \]  

(4)

where the tag-frequency for a tag \( t \) in \( T_{s_{ji}} \) is the number of times the tag was used to tag resources belonging to the cluster as follows:

\[ n^t_{s_{ji}} = |\{ r \in R \mid (t, r) \in I_u \land r \in s_{ji} \}| \]  

(5)

Figure 1 depicts an example of a hierarchical clustering solution showing user interests and associated tags in a hybrid user profile (concept labels were added for illustrative purposes only, but the algorithm do not extract category labels). It can be observed in the figure that the resources below the concept Operating Systems (OS) are divided into two clusters, one referring to Linux and other to Mac OSX, and the most frequent tags in each group clearly correspond to each of these groups.

4. Experimental Results

Experiments were performed using data collected from del.icio.us social bookmarking system. From this site we gathered the complete personomies of ten different users appearing in the main page, including their bookmarks and associated tags. The reduced number of users is due to the cost of creating profiles starting from the extracted personomies which requires analyzing the content of Web pages when running the clustering algorithm, in a real setting this task is performed gradually over time by agents. Table 1 summarizes the characteristics of the extracted personomies.

For a given user \( u \in U \) and a given resource \( r \in R \), a tag recommender system tries to find a set of tags \( \tilde{T} (u, r) \subseteq T \) for the user to annotate the resource [9]. In order to evaluate
tag recommendation using hybrid profiles, the bookmarks and corresponding tag assignments of each personomy were divided into a training set of approximately 80% of the total tagged bookmarks and a test set containing the remaining 20%. Hybrid user profiles were built using the training set, including the hierarchical description of concepts, whereas tag assignments of bookmarks in the test set were used to evaluate the approach. It is worth noticing that in a real scenario the hierarchy is learnt using the browsed pages instead of the ones belonging to the folksonomy.

The quality of a given list of top-\(N\) recommendations was evaluated considering the number of hits. This is, the number of tag assignments in the test set that were also present in the top-\(N\) recommended tags [3]. If \(N\) is the total number of recommendations, the hit-rate of the recommendation algorithm is computed as follows:

\[
\text{Hit-rate} = \frac{\text{number of hits}}{N} \tag{6}
\]

High values of hit-rate indicate that the algorithm was able to predict the assignments in the test sets of the corresponding users. The performance of tag recommendation was also evaluated based on the standard precision and recall metrics defined as follows:

\[
\text{recall} \left( \hat{T} (u, r) \right) = \frac{1}{|R_{\text{test}}|} \sum_{r \in R_{\text{test}}} \frac{|\text{tags} (u, r) \cap \hat{T} (u, r)|}{|\text{tags} (u, r)|} \tag{7}
\]

\[
\text{precision} \left( \hat{T} (u, r) \right) = \frac{1}{|R_{\text{test}}|} \sum_{r \in R_{\text{test}}} \frac{|\text{tags} (u, r) \cap \hat{T} (u, r)|}{|\hat{T} (u, r)|} \tag{8}
\]

where \(r\) is a resource to be tagged, \(\hat{T} (u, r)\) the set of recommended tags and \(\text{tags} (u, r)\) the set of real tags assigned by the user to the resource. F-measure was used to combine precision and recall values:

\[
\text{F-measure} \left( \hat{T} (u, r) \right) = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{9}
\]

Figure 2(a) shows the average precision, recall and hit-rate resulting of generating top-5 recommendations for the ten users according to variations of the similarity threshold described in the previous section. Precision increases as the similarity threshold grows, since clusters are smaller in size and recommendations are based on fewer, but highly similar resources. Conversely, recall tends to decrease since smaller clusters offer less tag diversity. The best values of hit-rate can be found in the interval \(0.1 \leq \delta \leq 0.3\), within which also the best relation between precision and recall is attained for most users. F-measure curves for the ten users are depicted in Figure 2(b) illustrating the relation between these two measures.

Hybrid profiles constructed using bookmarks from the personomies were compared with tag recommendation based on two different approaches commonly used in folksonomies: most popular tags by user (MPTU) and most popular tags by resource (MPTR). In the first approach, tags assigned by a given user are sorted according to their frequency of occurrence in the user resources and the top-\(N\) tags are in turn applied to make recommendations. Tag-based profiles consisting of a single vector of tags, as the ones described in Section 2, follow this approach although using different weighting schemes. The second tag recommendation approach is based on collective knowledge instead of personal one. In this approach the tags more frequently associated to a resource by members of the community are selected for recommendation.

Figure 3 depicts the hit-rate of top-5 recommendations using the three different approaches. It can be observed in this figure that recommendations based on hybrid profiles consistently reached higher hit-rates than the approaches based on tag popularity. In turn, MPTR reached better results than MPTU for most users. The differences in the performance of hybrid profiles with respect to MPTU and MPTR tested with a paired two-tailed t-test resulted statistical significant at a level of \(\alpha = 0.05\) with \(p\)-values 0.0119 and 0.0001 respectively.

### 5. Conclusions

The rapid adoption of collaborative tagging systems by diverse user communities on the Web provide novel sources of information for discovering user interests and preferences that can be exploited for recommendation. In this

![Table 1. Summary of the personomies obtained from del.icio.us](image-url)
paper, we proposed the integration of content-based profiles acquired by personal agents with data available in folksonomies about tag assignments done by users. Hybrid user profiles resulting from relating long-term user interests with tags frequently used in association to them can be applied to the recommendation of users, resources or tags in the context of folksonomies.

Experimental results of using hybrid profiles in tag recommendation for a number of users were presented, showing that these profiles are able to outperform the results of two commonly used recommendation methods based on tag popularity. However, more detailed experimentation needs to be done in order to confirm these preliminary findings. In future works additional criteria considered important in tag recommendation such as non-obviousness and discriminating power of tags [5] will be evaluated for recommendations obtained with hybrid profiles.

Figure 2. Performance metrics obtained in tag recommendation

Figure 3. Comparison of results using hybrid profiles and other recommendation approaches

Acknowledgments

This research has been partially supported by ANPCyT PICT N° 34917.

References


