Recommender Systems, Semantic-Based

Affiliation

Fatih Gedikli, Dietmar Jannach
Department of Computer Science, TU Dortmund, Dortmund, Germany
fatih.gedikli@tu-dortmund.de
dietmar.jannach@tu-dortmund.de

Synonyms

Social recommender systems, Web 2.0 recommender systems, Tag-based recommendation

Glossary

Collaborative filtering: a recommendation method which is based on rating information of the user community

Content-based filtering: a recommendation method which is based on characteristics of the recommended items as well as individual user feedback

Hybrid recommender system: a recommender system that combines different recommendation approaches or data sources

Rating matrix: a grid containing the users' implicit or explicit item ratings

Cold start problem: the ramp-up phase of a recommender where preference data is missing

Definition

Recommender Systems (RS) are software tools that are predominantly used on e-commerce sites and for other online services as a means to help the online customer find the most relevant shopping items or pieces of information quickly. Today, such systems can be found for a variety of different domains such as books, movies, music, hotels, restaurants, or news.

The particularity of RS is that they are able to provide personalized recommendations, which are based on the past and current behaviour or the explicit preferences of individual users, the preferences of a user community as a whole, or on various other forms of available information.

The main task of an RS usually is to predict as precisely as possible, which of the recommendable items will be of interest for and accepted by the user. Since the mid-1990s, when RS started to emerge as a research field of their own, a large variety of methods have been proposed to increase the quality of the recommendations, measured, e.g., in terms of their accuracy.

In their early years, RS were mainly considered as a tool for e-commerce sites or for personalized information filtering. However, the emergence of the Social Web – or, more generally, the Web 2.0 – soon had a strong impact on the field of RS. One aspect is related to the amount of information we know about the users, which is crucial for the recommendation quality. The Web 2.0 is participatory: People connect on social networks and share information about their personal profile and their interests, they actively contribute content on blogs and micro blogs and they share, rate, and review all types of resources online. Overall, much more information about the user is theoretically available than in the past, when explicit and implicit rating data and the past transaction history were often the only available knowledge sources.

Beside an increased engagement of users, the Web 2.0 also brought new application fields for RS technology. Today, we can find systems on Social Web platforms that recommend people to connect with or people to follow, systems that generate personalized information feeds based on the user's interests and systems that recommend potentially interesting Web resources such as images, web pages or blog posts. Even the choice of an appropriate set of tags and annotations for user-contributed content can be driven by an RS.

In this article, we will focus on RS that exploit (semantic) knowledge sources that have become available in the Social Web. In particular, we will focus on the role of user-provided tagging data in the recommendation process.

Introduction

With the continuously growing amount of information on the Web, the availability of appropriate tools that help the online user retrieve or discover interesting items becomes more and more important. Recommender systems are one type of such tools which are capable of generating personalized lists of shopping items, reading lists, or, more generally, action alternatives [1].

The recommendations of an RS can be based on different types of information. In most cases, the quality of the recommendations and the corresponding effect on the users are directly related to the amount and quality of the available information on which the recommendations are based on. Today, the most popular class of recommendation methods is called collaborative filtering (CF). CF methods rely on the existence of item ratings which are provided by an implicit online community. Amazon.com is an example of an online retailer who relies among others on such methods in their recommendation engines [2].

The other major type of systems is based on what is called "content-based filtering". While CF recommender systems recommend items similar users liked in the past, the task of a content-based recommender system is to recommend items that are similar to those the target user liked in the past.

We illustrate the basic rationale of a content-based recommendation method with an example from the movie domain. Table 1 represents an excerpt from an example movie database which also provides plot keywords for each movie, i.e., an item's content description is represented by a set of plot keywords. Table 2, on the other hand, shows an excerpt from the user database.

Movie	Plot Keywords			
Heat	Detective	**Criminal**	Thief	**Gangster**
Scarface	**Gangster**	**Criminal**	Drugs	Cocaine
Amélie	**Love**	Waitress	France	Happiness
Eat Pray Love	Divorce	India	**Love**	Inner Peace

Table 1: Movie data set with content description.

User	Preference Profile
Alice	Eat Pray Love, What Women Want
Bob	Scarface, Carlito's Way, Terminator II

Table 2: User database.

A simple content-based recommender computes recommendations for *Alice* by selecting movies Alice is not aware of and which are similar to those movies she watched before. In this example, similarity between movies could be defined by the number of overlapping keywords. The unseen movie *Amélie*, for example, has one keyword in common with *Eat Pray Love* ("Love"). Therefore, we can assume some degree of similarity between both movies. Since *Alice* liked *Eat Pray Love* in the past, the movie *Amélie* could be recommended to her.

Note that for a content-based recommender no user community is required for generating recommendations. However, the target user has to provide an initial list of "like" and "dislike" statements or ratings on a given scale. Alternatively, customer actions such as viewing or purchasing an item can be interpreted as positive signals. New items, on the other hand, can be incorporated in the recommendation process because similarity to existing items can be computed without the need for any rating data.

In the example discussed above the importance of each keyword was not taken into account, that is, each keyword gets the same importance. However, it appears intuitive that keywords which appear more often in descriptions are less representative. Therefore, the TF-IDF encoding format was proposed and gained popularity, in particular in the field of information retrieval, and is also the basis for various approaches that exploit Social Web data. TF-IDF stands for *term frequency – inverse document frequency* and is used to determine the relevance of terms in documents of a document collection. For convenience, we will assume in the following that the underlying item set consists of text documents, e.g., the plot keywords for each movie in Table 1 can be seen as one document. As the name suggests, the TF-IDF measure is composed of two frequency measures. The idea of the *term frequency* measure $TF(i,j)$ is to estimate the importance of a term i in a given document j by counting the number of times a given term i appears in document j. Additionally, a normalization is possible, e.g., by dividing the absolute number of occurrences of term i in document j by the absolute number of occurrences of the most frequent word in document j. Several other schemes are however possible.

On the other hand, the idea of the *inverse document frequency* measure $IDF(i)$ is to capture the importance of a term i in the whole set of available documents. Therefore, $IDF(i)$ can be seen as a global measure, which reduces the weight of words that appear in many documents (e.g., stop-words such as "a", "by", or "about"), since they are usually not representative and helpful to differentiate between documents. Formally, inverse document frequency is usually computed as $IDF(i) = \log\frac{N}{n(i)}$ where N is the size of the document set and $n(i)$ is the number of documents in which the given term i appears. We assume that each term appears in at least one document, i.e. $n(i) \geq 1$. If $n(i) = N$ the logarithm function returns 0 indicating that term i is of no importance for discriminating documents as it appears in all documents.

Finally, the TF-IDF measure, which represents the weight for a term i in document j, is defined as the combination of these two measures: $TF-IDF(i,j) = TF(i,j) * IDF(i)$.

With the help of the TF-IDF measure, text documents, or generally speaking the textual description of items, can be encoded as TF-IDF weight vectors.

One way of computing n recommendations is to find the n most similar items to the user's average TF-IDF weight vector of the user's liked documents. The cosine similarity metric is often used for computing the proximity between items.

Next, we will view user-provided tags as content descriptors and describe the role of tagging data in the recommendation process.

Recommendations based on Social Web tagging data

The advent of the Social Web opened new ways of promoting and sharing user-generated content. Web site visitors turned from passive recipients of information into active and engaged contributors. The Social Web allows users to create and share a large amount of different types of content such as pictures, videos, bookmarks, blogs, comments, or tagging data. It allows users to collaborate with other users on new types of Web applications called Social Web platforms such as Delicious [33] and Flickr [34]. Leveraging useful data from the large amount of user-contributed data available in the Social Web represents a challenging topic, which however also opens new opportunities for recommender system research.

For example, user-contributed tags are today a popular means for users to organize and retrieve items of interest in the Social Web. As the application areas of tags are manifold, they play an increasingly important role in the Social Web. They can be used to categorize items, express preferences about items, retrieve items of interest, and so on.

Collaborative tagging or social tagging describes the practice of collaboratively annotating items with freely chosen tags [3], which plays an important role in sharing content in the Social Web [4]. In a social tagging system such as Delicious and Flickr, users typically create new content (items), assign tags to these items, and share them with other users [5]. The result of social tagging is a complex network of interrelated users, items, and tags often referred to as a community-created *folksonomy*. The term folksonomy is a neologism introduced by the information architect Thomas Vander Wal [35] and is composed of the terms *folk* as in people and *taxonomy*, which stands for the practice and science of classification. A folksonomy is defined as a tuple $F := (U, T, R, Y)$ where U, T and R are finite sets, whose elements are called users, tags, and resources, and Y is a ternary relation between them, i.e., $Y \subseteq U \times T \times R$ called tag assignments.

In contrast to typical taxonomies including formal Semantic Web ontologies, social tagging represents a more light-weight approach, which does not rely on a pre-defined set of concepts and terms that can be used for annotation.

Tagging data also gained importance in the field of RS. User-generated tags not only convey additional information about the items, they also tell something about the user. For example, if two users use the same set of tags to describe an item, we can assume a certain degree of similarity between those. Therefore, tagging data can be used to augment the basic user-item rating matrix.

In the following, a possible categorization of building tag-based RS is given.

Using tags as content. Maybe the easiest way to use tagging data for RS is to consider tagging data as an additional source of content. Several works exist that view tags as content descriptors for content-based systems, see, for example, in [6, 7] or [8].

Similarly, in [9], tagging data is used for an existing content-based recommender system in order to increase the overall predictive accuracy of the system. Machine learning techniques are applied both on the textual descriptions of items (static data) and on the tagging data (dynamic data) to build user profiles and learn user interests. The user profile consists of three parts: the static content, the user's personal tags, and the social tags, which build the collaborative part of the user profile. Thus, in this work, tags are seen as an additional source of information used for learning the profile of a particular user. The authors compare their tag-based approach with a pure content-based recommender in a user study. The results show that the recommendations made by the tag-augmented recommender are slightly more accurate than the recommendations of the pure content-based one.

In [6], tags are also seen as content descriptors for different content-based systems. Tags are used for building user profiles for the popular music community site Last.fm. To address the so-called cold start problem (when new users or items enter in the system), the user profiles are inferred automatically, e.g., from the music tracks available on the computer of each user, thus reducing the manual effort from the user's side to express his or her preferences. The authors show that tag-based profiles can lead to better music recommendations than conventional user profiles based on song and track usage.

In [5], tags are considered as content features that describe both user and item profiles. The authors propose weighting functions which assess the importance of a particular tag for a given user or item, and similarity functions which compute the similarity between a user profile and an item profile. These weighting and similarity functions are then combined in different content-based recommendation models.

In that work, user interests and item characteristics are modeled as vectors $u_m = (u_{m,1}, \ldots, u_{m,L})$ and $i_n = (u_{n,1}, \ldots, u_{n,L})$ of length L respectively, where L is the number of tags in the folksonomy, $u_{m,l}$ is the number of times user u_m has annotated items with tag t_l, and $i_{n,l}$ is the number of times item i_n has been annotated with tag t_l. After modeling users and items as vectors accordingly, the authors can adopt the TF-IDF vector space model.

The evaluation results on the Delicious and Last.fm data sets show that the recommendation models focusing on user profiles outperform the models focusing on item profiles.

Tagging data can also be incorporated in search engines to personalize the search results. According to [10], two basic approaches to Web search personalization can be differentiated. In the first approach, a user's original query is modified and adapted to the profile of the user. For example, the query "eclipse" might be extended to "eclipse software development environment" if we know that the user has an interest in software development. In the second approach, the query is not modified, but the returned list of search results is re-ranked according to the user profile.

An example for the latter approach is given in [30]. The authors propose a pure tag-based personalization method to re-rank the Web search results, which is independent from the underlying search engine. The basic idea is to use bookmarks and tagging data to re-rank the documents in the search result list. The authors propose a concept called *tagmarking*, which translates the keywords in the search query to tags and assign them to the bookmarked Web page that is associated with the query. Bookmarks and tags are aggregated in a binary *tag-document* matrix where each column (vector) represents a bookmark of a document with its components set to 1 when the corresponding tag is associated with the document and 0 otherwise. The user profile is modeled as a vector which contains the weights assigned to each tag. The *tag-user* matrix and the document profile are built analogously. Finally, in the personalization step, the documents are re-ranked according to a similarity metric which combines both the user profile and the document profile. Table 3 shows in an example of how personalization affects Google's result list for the search query "security", see also [30]. The ranking of the Web site of the US Social Security Administration (ssa.gov), for instance, has increased because – according to the authors – the user who submitted the query also showed interest in insurance matters.

Rank	Δ Rank	URL
1	•	securityfocus.com
2	↑ +7	cert.org
3	•	microsoft.com/technet/security/def...
4	↑ +4	w3.org/Security
5	↑ +2	ssa.gov
6	↑ +4	nsa.gov
7	↓ −5	microsoft.com/security
8	↓ −2	windowsitpro.com/WindowsSecurity
9	↓ −4	whitehouse.gov/homeland
10	↓ −6	dhs.gov

Table 3: Re-ranking Google's result list [30].

Clustering approaches. Many tag-based clustering approaches have been proposed in the literature which cluster users and items according to topics of interest by exploiting additional tagging data, see for example [7, 11] or [12].

In [7], the authors propose a system called *Internet Social Interest Discovery* (ISID) and show its application for the social bookmarking system Delicious. The ISID system, as the name suggests, is a system specifically designed to reveal common user interests based on user-provided tags. The basic assumption, which is then justified in the work, is that user-provided tags are more effective at reflecting the users' understanding of the content than the most-informative keywords extracted from the corpus of a Web page. Therefore, tags are seen as good candidates for capturing user interests.

Similarly, in [11], co-occurring tags are used to build topics of interests. In the resource-tag matrix, each tag is described by a set of resources to which this tag has been assigned. Afterwards, the authors obtain the tag similarity matrix by computing the cosine similarity between the tag vectors in the resource-tag matrix. Based on this similarity matrix, a graph is constructed where the tags represent the nodes and the edges represent the similarity relationships between the tags. Afterwards, a clustering algorithm is used to cluster the tags and to extract the topics of interests. Finally, the authors present the topic-oriented tag-based recommendation system *TOAST*. TOAST applies preference propagation on an undirected graph called the "topic-oriented graph" which consists of three kinds of nodes: users, resources, and topics. In their recommendation strategy, the authors propagate a user’s preference through transitional nodes such as users, resources, and topics, to reach an unknown resource node along the shortest connecting path.

In [13], the authors focus on a recommendation scenario where a user selects a tag and expects a recommendation of related resources. They thus present a recommendation approach which recommends items for a given user-tag pair (u,t). Tag clusters are presumed to act as a bridge between users and items. The idea behind tag clusters is to account for the effects of unsupervised tagging such as redundancy and ambiguity. The authors first determine the items which have some similarity to the query tag t. These items are then re-ranked according to the user profile. The ranking algorithm first calculates the user's interest with respect to each tag cluster as well as the nearest clusters of each item. The nearest clusters are determined by counting the number of times the item was annotated with a tag from the cluster divided by the total number of times the item was annotated. Both measures are then combined in the final personalized rank score used to re-rank the item sets. The results show that data sparsity has a big influence on the quality of the clusters which, on the other hand, corresponds with the accuracy of the recommendations.

Hybrid approaches. Hybrid approaches in general combine different sources of information or different algorithms to make recommendations. In the context of semantic-based recommenders, social data such as tagging data can be combined with other types of information such as content data [14] or data from the Semantic Web [15].

In [14], a Bayesian model-based recommender that leverages content and social data is presented. In [15], on the other hand, a tag-based recommender which recommends Web pages is extended such that also semantic similarities between tags are discovered, which are usually not taken into account in syntax-based similarity approaches. Consider the example in Table 4.

Web page	Tags
P1	Programming, Web 2.0, Framework
P2	PHP, Scripting, Web 2.0
P3	C++, Programming, Framework

Table 4: Exploiting semantic relations between tags.

If we assume a syntax-based similarity measure, the Web pages *P1* and *P3* will be considered more similar than *P1* and *P2* as *P1* and *P3* have two tags in common ("Programming" and "Framework"), whereas *P1* and *P2* only share one tag ("Web 2.0"). However, if we analyze the tags in more detail, we see that *P1* is closer to *P2* than to *P3* because *P1* and *P2* are about Web technologies, whereas *P3* focuses on C++, which is a programming language that is usually not associated with Web technologies. In a semantic-based similarity approach which takes lexical and social factors of tags into account, these semantic relations can be made explicit. For example, "Web 2.0" would be considered together with "Scripting", and "Programming" together with "PHP". The authors try to overcome this problem of ignoring the semantic term relations by hybridizing syntax-based approaches such as tag popularity with a new semantic-based approach. In particular, they also make use of external semantic sources such as the WordNet dictionary and different ontologies from Open Linked Data available on the Web to identify semantic relations between tags. These semantic relations are then considered in the similarity calculations. Their experimental results show increases of precision when semantic relations are exploited as additional knowledge sources.

Tag-enhanced collaborative filtering. A substantial number of papers have been published in recent years on tag-enhanced CF recommender algorithms in which tagging data is used for improving the performance of traditional collaborative filtering recommender systems. In general, tagging data can be incorporated into existing collaborative filtering algorithms in different ways in order to enhance the quality of recommendations [4, 17] or [23].

In [17], for example, the authors incorporate tags into standard collaborative filtering algorithms. The idea is to reduce the three-dimensional relation $\langle user, item, tag \rangle$ to three two-dimensional relations, namely $\langle user, tag \rangle$, $\langle item, tag \rangle$, and $\langle user, item \rangle$. The projection is based on viewing the tags as items ("user tags") and users ("item tags") respectively. For example, in the $\langle user, tag \rangle$ relation, tags are viewed as items in the user-item rating matrix. These so called user tags represent tags that are used by the users to tag items. On the other hand, item tags in the $\langle item, tag \rangle$ relation correspond to tags that describe the items. Considering the ternary relation as three two-dimensional relations enables the authors to apply standard collaborative filtering techniques. The authors also propose a fusion method which re-combines the individual relations. The results of their empirical analysis show that the predictive performance of their proposed fusion method which incorporates tags outperforms the standard tag-unaware collaborative filtering algorithms.

Exploiting tagging data without reducing the three-dimensional $\langle user, item, tag \rangle$ relation was the next logical step. In recent years, recommendation methods were proposed which can directly exploit the ternary relationship in tagging data [18, 19, and 20].

In [21], the authors present a *graph-based* tag recommender algorithm called *FolkRank*. As the name suggests, the FolkRank algorithm is based on Google's PageRank algorithm. The main idea of PageRank is that pages are important when linked by other important pages. Therefore, PageRank views the Web as a graph and uses a weight spreading algorithm to calculate the importance of the pages. FolkRank adopts this idea and assumes that a resource is important if it is tagged with important tags from important users.

A major problem of FolkRank is that it does not scale to larger problem sizes, which is crucial for real-world scenarios. Therefore, in [27], *LocalRank* – a graph-and-neighborhood-based tag recommendation approach – is presented. Rank computation and weight propagation in LocalRank is done in a similar way to FolkRank but without iterations. As the name suggests, LocalRank computes the rank weights based only on the local "neighborhood" of a given user and resource. Unlike the FolkRank algorithm which considers all elements in the folksonomy, LocalRank focuses on the relevant ones only. Thus, LocalRank can significantly reduce the time needed for computing the recommendations while maintaining or slightly improving recommendation quality.

Tensor Factorization (TF) represents another method to directly exploit the ternary relationship in tagging data. In [19], the authors see the ternary relationship as a three dimensional tensor (cube) and apply the idea of computing low rank approximations for tensors on a tag recommender algorithm. The evaluation results show that their TF-based method achieves even better accuracy results than the tag recommender algorithm FolkRank [21]. However, the TF-based model comes with the problem of a cubic runtime in the factorization dimension for prediction and learning. This problem is addressed in the work of [20]. The authors present a *Pairwise Interaction Tensor Factorization* (PITF) model with a linear runtime in the factorization dimension. The PITF model explicitly models the pairwise interactions between users, items, and tags.

In [22], the authors propose tag-based recommender algorithms which they call "tagommenders". The idea is to utilize *tag preference data* in the recommendation process in order to generate better recommendation rankings than state-of-the-art baseline algorithms. The authors define a user's preference for a tag as the user's level of interest in items, e.g. movies, exhibiting the concept represented by the tag. Thus, a user can, for example, indicate that he or she likes *animated* movies, but dislikes movies about *serial killers*. However, since no tag preference data is available, the tag preferences of the target user have to be estimated before the algorithm can predict a user's preference for the target item. To that purpose, the authors evaluate a variety of tag preference inference algorithms. Such algorithms estimate the user's attitude toward a tag, that is, if and to which extent a user likes items that are annotated with a particular tag. Their results show that a linear combination of all preference inference algorithms performed best, that is, algorithms that exploit a variety of signals such as implicit and explicit user data work best.

The proposed tagommender algorithms rely on "global" tag preferences: a tag is either liked or disliked by a user, independent of a specific item. In contrast, in [26] and [31] the concept of *item-specific* tag preference data was introduced. The intuition behind this idea is that the same tag may have a positive connotation for the user in one context and a negative in another. For example, a user might like *action movies* featuring the actor *Bruce Willis*, but at the same time the user might dislike the performance of *Bruce Willis* in *romantic movies*. Based on such an approach, users are able to evaluate an item in various dimensions and are thus not limited to the one single overall vote anymore. According to the study presented in [31], users particularly appreciated this new feature, a fact that was measured in increased user satisfaction. In [26], the authors present first recommendation schemes that take item-specific tag preferences into account when generating rating predictions. The results show that the accuracy can be further improved by exploiting item-specific tag preference data.

Tag-based explanations. Tagging data is not only a means to enhance existing recommender algorithms but it can also serve as a means to strengthen and improve explanations for recommendations. Explanations are one of the current research topics in recommender system research. They play an increasingly important role as they can significantly influence the way a user perceives the system.

In [24], tag-based explanation interfaces, which the authors call "tagsplanations", are described and evaluated. The authors propose explanation interfaces which use *tag relevance* and *tag preference* as two key components. Tag relevance measures the strength of the relationship of the tag to the item, while tag preference indicates the strength of the relationship between a user and the tag. Consider, for example, the tag "love" for a given user–item (movie) pair. Tag preference measures how well the tag "love" describes the particular movie, while tag preference indicates the user's interest in movies about love, that is, how much the user likes/dislikes movies about love in general, independent from a particular movie.

In [25], the authors introduce explanation interfaces based on personalized and non-personalized tag clouds. They compare tag cloud based explanations with keyword style explanations proposed in previous work. In order to personalize the explanations, the personalized tag cloud interface makes use of tag preference data proposed in [26]. These item-specific tag preference values are then mapped to colors which indicate whether the user will like, dislike, or feel neutral about the item features represented by the tags in the cloud. In the example tag cloud in Figure 1, blue is used as a color for users, for which the system knows or assumes that the user has positive feelings about, e.g., the tag "family". Red tags such as "divorce", on the other hand, represent aspects the user will probably not like. Tags which are marked as neutral are printed in black. The results of their user study showed that users can make better decisions faster when using the tag cloud interfaces rather than the keyword-style explanations. In addition, users generally favored the tag cloud interfaces over keyword-style explanations.

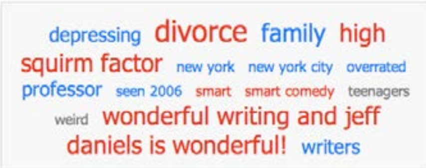

Figure 1: Personalized Tag Cloud Explanation.

Perspectives

In recent years, exploiting tagging data for recommendations has become an active research topic in the field of RS. Tag-based computing can further improve the quality of RS and leads to new possibilities but also to a number of new research questions. For example, opinion mining based on folksonomies represents one challenging topic which is currently being addressed in literature. The task of opinion mining is to extract the users' sentimental orientations or attitudes to items based on different information sources such as reviews, blogs, and comments. Recently, user-provided tags are recognized as one such information pool as the tagging of items also tells something about the user. The hybridization of these information sources also plays an increasingly important role. In [32], for example, the authors combine a user-provided folksonomy and an expert-driven taxonomy to assess a user's opinion about an item and to make personalized recommendations. They show that by taking the expert's viewpoint into account, the accuracy of item recommendations can be further improved. Future work might aim to integrate tagging data with further information sources such as reviews or blogs.

Furthermore, we believe that future work will concentrate on topics of bringing semantics to tagging data (see, for example, [16] and [28]). Semantically enhanced tags will further improve various aspects of recommender systems such as accuracy, diversity, or explanation facility.

In general, we see tagging data as a promising source of information to further improve different technologies and approaches related to the Semantic Web and the Social Web [29] and in particular to develop more powerful applications for search and recommendation [30].

Cross-References

- Analysis and Mining of Tags, (Micro-)Blogs, and Virtual Communities (00172)
- Folksonomies (00110)
- Human Behavior and Social Networks (00235)
- Recommender Systems, Semantic based (00116)
- Tag Clouds (00126)

References

1. Jannach, D., Zanker, M., Felfernig, A., Friedrich, G.: Recommender Systems - An Introduction, *Cambridge University Press*, 2010.
2. Linden, G. Smith, and J. York. Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Computing, 7(1):76–80, 2003.
3. Golder, S. A., and Huberman, B. A.: Usage patterns of collaborative tagging systems. Journal of Information Science, 32(2):198–208, 2006.
4. Ji, A.-T., Yeon, C., Kim, H.-N., and Jo, G.-S.: Collaborative tagging in recommender systems. In: AUS-AI'07, pp. 377–386, 2007, Gold Coast.
5. Cantador, I., Bellogin, A., and Vallet, D.: Content-based recommendation in social tagging systems. In: RecSys'10, pp. 237–240, 2010, Barcelona.
6. Firan, C. S., Nejdl, W., and Paiu, R.: The benefit of using tag-based profiles. In: LA-WEB'07, pp. 32–41, 2007, Santiago de Chile.
7. Li, X., Guo, L., and Zhao, Y. E.: Tag-based social interest discovery. In: WWW'08, pp. 675–684, 2008, Beijing.
8. Vatturi, P. K., Geyer, W., Dugan, C., Muller, M., and Brownholtz, B.: Tag-based filtering for personalized bookmark recommendations. In: CIKM'08, pp. 1395–1396, 2008, Napa Valley.
9. de Gemmis, M., Lops, P., Semeraro, G., and Basile, P.: Integrating tags in a semantic content-based recommender. In: RecSys'08, pp. 163–170, 2008, Lausanne.
10. Pitkow, J., Schütze, H., Cass, T., Cooley, R., Turnbull, D., Edmonds, A., Adar, E., and Breuel, T.: Personalized search. Communications of the ACM, 45(9):50–55, 2002.
11. Xu, G., Gu, Y., Zhang, Y., Yang, Z., and Kitsuregawa, M.: Toast: A topic-oriented tag-based recommender system. In: WISE'11, pp. 158–171, 2011, Sydney.
12. Zanardi, V. and Capra, L.: A scalable tag-based recommender system for new users of the Social Web. In: DEXA'11, pp. 542–557, 2011, Toulouse.
13. Shepitsen, A., Gemmell, J., Mobasher, B., and Burke, R.: Personalized recommendation in social tagging systems using hierarchical clustering. In: RecSys'08, pp. 259–266, 2008, Lausanne.
14. Seth, A. and Zhang, J.: A social network based approach to personalized recommendation of participatory media content. In: ICWSM'08, 2008, Seattle.
15. Durao, F. and Dolog, P.: Extending a hybrid tag-based recommender system with personalization. In: SAC'10, pp. 1723–1727, 2010, Sierre.
16. Xu, G., Gu, Y., Dolog, P., Zhang, Y., and Kitsuregawa, M.: Semrec: A semantic enhancement framework for tag

based recommendation. In: AAAI'11, pp. 1267–1272, 2011, San Francisco.
17. Tso-Sutter, K. H. L., Marinho, L. B., and Schmidt-Thieme, L.: Tag-aware recommender systems by fusion of collaborative filtering algorithms. In: SAC'08, pp. 1995–1999, 2008, Fortaleza.
18. Symeonidis, P., Nanopoulos, A., and Manolopoulos, Y.: Tag recommendations based on tensor dimensionality reduction. In: RecSys'08, pp. 43–50, 2008, Lausanne.
19. Rendle, S., Balby Marinho, L., Nanopoulos, A., and Lars, S.-T.: Learning optimal ranking with tensor factorization for tag recommendation. In: SIGKDD'09, pp. 727–736, 2009, Paris.
20. Rendle, S. and Schmidt-Thieme, L.: Pairwise interaction tensor factorization for personalized tag recommendation. In: WSDM'10, pp. 81–90, 2010, New York.
21. Hotho, A., Jäschke, R., Schmitz, C., and Stumme, G.: Information retrieval in folksonomies: Search and ranking. In: ESWC'06, pp. 411–426, 2006, Budva.
22. Sen, S., Vig, J., and Riedl, J. T.: Tagommenders: Connecting users to items through tags. In: WWW'09, pp. 671–680, 2009, Madrid.
23. Zhen, Y., Li, W.-J., and Yeung, D.-Y. (2009). Tagicofi: Tag informed collaborative filtering. In: RecSys'09, pp. 69–76, 2009, New York.
24. Vig, J., Sen, S., and Riedl, J. T.: Tagsplanations: Explaining recommendations using tags. In: IUI'09, pp. 47–56, 2009, Sanibel Island.
25. Gedikli, F., Ge, M., and Jannach, D.: Understanding recommendations by reading the clouds. In: EC-Web'11, pp. 196–208, 2011, Toulouse.
26. Gedikli, F. and Jannach, D.: Improving recommendation accuracy based on item-specific tag preferences. ACM Transactions on Intelligent Systems and Technology, 4(1) 2013.
27. Kubatz, M., Gedikli, F., and Jannach, D.: LocalRank - Neighborhood-based, fast computation of tag recommendations. In: EC-Web'11, pp. 258–269, 2011, Toulouse.
28. Cattuto, C., Benz, D., Hotho, A., and Stumme, G.: Semantic grounding of tag relatedness in social bookmarking systems. In: ISWC'08, pp. 615–631, 2008, Karlsruhe.
29. Passant, A.: Using ontologies to strengthen folksonomies and enrich information retrieval in weblogs. In: ICWSM'07, 2007, Boulder.
30. Noll, M. G. and Meinel, C.: Web search personalization via social bookmarking and tagging. In: ISWC'07/ASWC'07, pp. 367–380, 2007, Busan.
31. Vig, J., Soukup, M., Sen, S., and Riedl, J. T.: Tag expression: Tagging with feeling. In: UIST'10, pp. 323–332, 2010, New York.
32. Liang, H., Xu, Y., and Li, Y.: Mining users' opinions based on item folksonomy and taxonomy for personalized recommender systems. In: ICDM'10, 2012, Sydney.
33. http://www.delicious.com
34. http://www.flickr.com
35. http://vanderwal.net/folksonomy.html

Recommended Reading

1. Ricci, F.; Rokach, L.; Shapira, B.; Kantor, P.B. (Eds.), Recommender Systems Handbook. Springer, 2011.
2. Jannach, D., Zanker, M., Felfernig, A., Friedrich, G.: Recommender Systems - An Introduction, Cambridge University Press, 2010.