Collaborative Tag Recommendations

Leandro Balby Marinho and Lars Schmidt-Thieme

Information Systems and Machine Learning Lab (ISMLL)
Samelsonplatz 1, University of Hildesheim, D-31141 Hildesheim, Germany
{marinho,schmidt-thieme}@ismll.uni-hildesheim.de

Abstract. With the increasing popularity of collaborative tagging systems, services that assist the user in the task of tagging, such as tag recommenders, are more and more required. Being the scenario similar to traditional recommender systems where nearest neighbor algorithms, better known as collaborative filtering, were extensively and successfully applied, the application of the same methods to the problem of tag recommendation seems to be a natural way to follow. However, it is necessary to take into consideration some particularities of these systems, such as the absence of ratings and the fact that two entity types in a rating scale correspond to three top level entity types, i.e., user, resources and tags. In this paper we cast the tag recommendation problem into a collaborative filtering perspective and starting from a view on the plain recommendation task without attributes, we make a ground evaluation comparing different tag recommender algorithms on real data.

1 Introduction

The process of building the Semantic Web (Berners-Lee et al. 2001) is currently an area of high activity. Both the theory and technology to support it have been already defined and now one must fill this structure with life. In spite of the sounding simplicity, this task actually represents the biggest challenge towards its realization, i.e., adding semantic annotation to Web documents and resources in order to provide knowledge access instead of unstructured material. Annotation represents an extra effort which certainly will not be voluntarily done without good reasons. In this sense, it is necessary to incentive and educate the user into this practice, e.g., showing the benefits that can be achieved through it and alleviating the extra burden with the recommendation of relevant annotations. With the recent appearing and increasing popularity of the so called collaborative tagging systems this is finally possible (Golber et al. (2005)).

Recommending tags can serve various purposes, such as: increasing the chances of getting a resource annotated (or tagged) and reminding a user what a resource is about. Furthermore, lazy annotating users would not need to come up with a tag themselves but just select the ones readily available in the recommendation list according to what they think is more suitable for the given resource.

Tag recommender systems recommend relevant tags for an untagged user resource. Relevant here can assume different perspectives, for example, a tag can be judged relevant to a given resource according to the society point of view, through the opinion of experts in the domain or even based on the personal profile of an individual user. The question would be, which concept of relevance would the user prefer the most when using tag recommender services. This paper attempts to address this question through the following contributions: (i) formulation of the tag recommendation problem and the introduction of a collaborative filtering-based tag recommender algorithm, (ii) presentation of a simple protocol for tag recommender evaluation (iii) and (iv) a ground and quantitative evaluation on real-life data comparing different tag recommender algorithms.

2 Related work

The literature regarding the specific problem of collaborative tag recommendation is still sparse. The majority of the recent research work about collaborative tagging systems and folksonomies is concerned in devising approaches to better structure the data for browsing and searching where the recommendation problem is sometimes only highlighted as a potential property to be further explored in future work (Mika (2005), Hotho et al. (2006), Brooks and Montanez (2006), Heymann and Garcia-Molinay (2006)). We briefly describe below the works specifically investigating the problem of collaborative tag recommendation.

Autotag (Mishne (2006)) is a tool that suggests tags for weblog posts using collaborative filtering methods. Given a new weblog post, posts which are similar to it are identified through traditional information retrieval similarity measures. Next, the tags assigned to these posts are aggregated creating a ranked list of likely tags. Despite the collaborative filtering scenario, there is no real personalization because the user is not taken directly into account. Furthermore, the evaluation is done in a semi-automatically fashion where the assumption of tag relevance for a given resource is defined to some extent by human experts.

Xu et al. (2006) introduce a collaborative tag suggestion algorithm based on a set of general criteria to identify high quality tags. Some of the considered criteria are: high coverage of multiple facets to ensure good recall, least effort to reduce the cost involved in browsing, and high popularity to ensure tag quality. A goodness measure for tags, derived from collective user authorities, is iteratively adjusted by a reward-penalty algorithm, which also incorporates other sources of tags, e.g., content-based auto-generated tags. There is no quantitative evaluation.

Benz et al. (Benz et al. (2006)) introduce a collaborative approach for bookmark classification based on a combination of nearest-neighbor-classifiers. Two separate kinds of recommendations are generated: Keyword recommendations on the one hand, i.e. which keywords to use for annotating a new bookmark, and a recommendation of a classification on the other hand. The keyword recommender can be regarded as a collaborative tag recommender but its just a component of the overall algorithm, and therefore there is no information about its effectiveness as a standalone tool.

The state-of-the-art tag recommenders in practice are services that provide the most-popular tags used by the society for a particular resource (Fig. 2). This is usually done by means of tag clouds where the most frequently used tags are depicted in a larger font or otherwise emphasized.

The approaches described above address important aspects of the problem, but there is still a lack regarding quantitative evaluation on basic tag recommender algorithms. Furthermore, there is no common or agreed protocol where the different algorithms should be compared.

3 Recommender Systems

Recommender systems (RS) recommend products to customers based on ratings or past customer behavior. In general, RS predict ratings of items or suggest a list of unknown items to the user. They usually take the users, items and the ratings of items into account. A recommender system can be briefly formulated as:

- A set of users U
- A set of items I
- A set $\mathrm{S} \subseteq \mathbb{R}$ of possible ratings where $r : U \times I \to \mathrm{S}$ is a partial function that associates ratings to user/item pairs. In datasets r typically is represented as a list of tuples $(u, i, r(u, i))$ with $u \in U$, $i \in I$ and r defined for the domain $dom_r \subseteq U \times I$
- Task: In recommender systems the recommendations are for a given user $u \in U$ a set $\tilde{I}(u) \subseteq I$ of items. Usually $\tilde{I}(u)$ is computed by first generating a ranking on the set of items according to some quality or relevance criterion, from which then the top n elements are selected (see Eq. 2 below).

In CF, for m users and n items, the user profiles are represented in a user-item matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$. The matrix can be decomposed into row vectors:

$$\mathbf{X} := [\mathbf{x}_1, ..., \mathbf{x}_m]^\top \text{ with } \mathbf{x}_u := [x_{u,1}, ..., x_{u,n}]^\top, \text{ for } u := 1, \ldots, m,$$

where $x_{u,i}$ indicates that user u rated item i by $x_{u,i} \in \mathbb{R}$. Each row vector $\mathbf{x}_u$ corresponds thus to a user profile representing the item ratings of a particular user. This decomposition leads to user-based CF.

The matrix can alternatively be represented by its column vectors:

$$\mathbf{X} := [\mathbf{x}_1, ..., \mathbf{x}_m] \text{ with } \mathbf{x}_i := [x_{i,1}, ..., x_{i,m}]^\top, \text{ for } i := 1, \ldots, n,$$

where each column vector $\mathbf{x}_i$ corresponds to a specific item's ratings by all m users. This representation leads to item-based recommendation algorithms.

The pairwise similarities between users is usually computed by means of vector similarity:

$$sim(\mathrm{prof}_u, \mathrm{prof}_v) := \frac{\langle \mathrm{prof}_u, \mathrm{prof}_v \rangle}{\| \mathrm{prof}_u \| \| \mathrm{prof}_v \|} \tag{1}$$

where $u, v \in U$ are two users and $prof_u$ and $prof_v$ are their profile vectors.

Let $B \subseteq I$ be the basket of items of the active user $u \subseteq U$ and N_u his/her best-neighbors. The topN recommendations usually consists of a list of items ranked by decreasing frequency of occurrence in the ratings of the neighbors:

$$\tilde{I}(u) := \underset{i \in I}{\arg\overset{n}{\max}} |\{v \in N_u \mid i \in r_{v,i}\}| \tag{2}$$

where $B \cap \tilde{I}(u) := \emptyset$ and n is the size of the recommendation list.

The brief discussion above refers only to the user-based CF case, since it is the focus of our work. Moreover, we consider only the recommendation task since in collaborative tagging systems there are no ratings and therefore no prediction. For a detailed description about the item-based CF algorithm see Deshpande et al. (2004).

4 Tag Recommender Systems

Tag recommender systems recommend relevant tags for a given resource. As already discussed in section 1, the notion of relevance here can assume different perspectives and is usually hard to judge what concept of relevance would be preferable to a particular user. Collaborative tagging systems usually allow the users to see the most popular tags used for a given resource. This can be thought of a social-based tag recommender service since it represents the society opinion as a whole. Through CF we can measure the extent to which personalized notions of tag relevance are preferable in comparison with the socialized ones.

Collaborative tagging systems are usually composed of users, resources and tags and allow users to assign tags to resources. What is considered a resource depends on the type of the system, e.g. URLs (del.icio.us[1]), pictures (Flickr[2]), music(Last.fm[3]), etc. A tag recommender system can be formulated as follows:

- A set of users U
- A set of resources R
- A set of tags T
- A function $s : U \times R \rightarrow \tilde{T}$ associating tags to user/resources pairs, where $\tilde{T} \subseteq T$ and s is defined for the domain $dom_s \subseteq U \times R$
- Task: In tag recommender systems the recommendations are for a given user $u \in U$ and a resource $r \in R$ a set $\tilde{T}(u,r) \subseteq T$ of tags. As well as in the traditional formulation (section 3), $\tilde{T}(u,r)$ can also be computed by first generating a ranking on the set of tags according to some quality or relevance criterion, from which then the top n elements are selected (see Algo.1 below).

When comparing the formulation above with the one in section 3, we observe that CF cannot be applied directly. This is due to the additional dimension represented by

[1] http://del.icio.us
[2] http://www.flickr.com
[3] http://www.last.fm

T. Either we use more complex methods do deal directly with it or reduce it to a lower dimensional space where we could apply CF. We follow the latter one.

To this end we take all the two dimensional projections of the original matrix preserving the user information. Letting $K := |U|$, $M := |I|$ and $L := |T|$, the projections result in two user profile matrices: a user-resource $K \times M$ matrix $\mathbf{X}$ and a user-tag $K \times L$ matrix $\mathbf{Y}$. In collaborative tagging systems there is usually no rating information. The only information available is whether or not a resource and/or a tag occurred with the user. This can be encoded in the binary matrices $\mathbf{X} \in \{0,1\}^{k \times m}$ and $\mathbf{Y} \in \{0,1\}^{k \times l}$ indicating occurrence, e.g. $x_{k,m} = 1$ and $y_{k,l} = 1$, or non-occurrence of resources and tags with the users. Now we have the required setup to apply collaborative filtering.

The algorithm starts selecting the users who have tagged the resource in question. Next, the pairwise similarity computation is performed (Eq.1). Notice that now we have two possible setups in which the neighborhood can be formed, either based on the profile matrix $\mathbf{X}$ or $\mathbf{Y}$. The neighborhood's tags for the resource in question are aggregated and weighted based on the neighbors' similarities with the active user. Next the weights of each particular tag are summed up and the recommendation list is ranked by decreasing value of the summed weights. Ties are broken by smaller index. The overall CF procedure for tag recommendations is summarized in Algo.1.

Algorithm 1 CF for tag recommendations

- Given a new and/or untagged resource $r \in R$ for the active user $u \in U$
- Let A $:= \{v \subseteq U \mid s_{v,r} \not\equiv \emptyset\}$ denote the set of users who have tagged r where s is a function associating tags to user/resources pairs
  - Find k best neighbors:
  $$N_u := \arg\max_{v \in A}^{k} sim(\text{prof}_u, \text{prof}_v)$$
  - Output the top n tags:
  $$\tilde{T}(u,r) := \arg\max_{t \in T}^{n} \sum_{v \in N_u} sim(\text{prof}_u, \text{prof}_v)\delta(v,r,t)$$
  where $\delta(v,r,t) := 1$ if $(v,r,t) \in U \times R \times T$ and 0 else.

5 Experimental setup and results

For our experiments we used the data made available by the Audioscrobbler[4] system, a music engine based on a collection of music profiles. These profiles are built through the use of the company's flagship product, Last.fm, a system that provides personalized radio stations for its users and updates their profiles using the music they listen to and also makes personalized artist recommendations. In addition, Audioscrobbler exposes large portions of data through their web services API.

[4] http://www.audioscrobbler.net

Fig. 1. Most popular tags for a given artist

Here we considered only the resources with 10 or more tag assignments. This gave us 2.917 users, 1.853 artists (playing the role of resources), 2.045 tags and 219.702 instances ((user, resource, tag) triples).

We evaluated four tag recommenders: (i) a *most global frequent tags*, which recommend the most used tags in the sample dataset, (ii) a *most popular tag by resource*, which recommends the most used tags for a particular resource (in our case an artist), (iii) a *user-resource-based CF*, which computes the neighborhood based on the user-resource matrix and (iv) a *user-tag-based CF*, which computes the neighborhood based on the user-tag matrix. Notice that (ii) represents the state-of-the-art recommender used in practice (Fig.1).

To evaluate the recommenders we used a variant of the leave-one-out holdout estimation that we named leave-tags-out. The idea is to choose a resource at random for each user in the test set and hide the tags attached to it. The algorithm must try to predict the hidden tags. To count the hits made by the algorithms we used the usual recall measure,

$$recall^{macro}(\mathcal{D}) := \frac{1}{\mid \mathcal{D} \mid} \sum_{1=1}^{|\mathcal{D}|} \frac{\mid Y_i \cap Z_i \mid}{\mid Y_i \mid} \tag{3}$$

where $\mathcal{D}$ is the test set, Y_i the true tags and Z_i the predicted ones. Since the precision is forced by taking into account only a restricted number n of recommendations there is no need to evaluate precision or F1 measures, i.e., for this kind of scenario precision is just the same as recall up to a multiplicative constant. Each algorithm was evaluated 10 times for n=10 (size of recommendation list) and the results averaged (Fig. 2).

Looking at the Figure 2 we see that the *most popular by resource* recommender reached a surprisingly high recall and that the *user-resource-based CF* did not perform significantly better than that. The good results of the *most popular by resource* algorithm can in part be explained by the fact that this service is already available by

Fig. 3. Recall for n varying from 1 to 10

Fig. 2. Recall of tag recommenders for n=10

the system. Besides that, it shows the strong influence of the society's vocabulary on the user's personal opinion. In the other hand, the *user-tag-based CF* recommender performed at least 2% better[5] than both the *most-popular tag by resource* and *user-resource-based CF*. Also notice that the improvement is consistent for different values of n (Fig. 3). The best k-neighbors values were estimated through successive runnings where k was incremented until a point where no more improvements in the results were observed.

6 Conclusions

In this paper we applied CF to the tag recommendation problem and made a quantitative evaluation of its performance in comparison with other simpler tag recommenders. Furthermore, we used a simple and suitable protocol with which further approaches can be compared.

Despite the already good results of the baseline algorithms, the straightforward CF based on the user-tag profile matrix showed a significant improvement. This shows that users with similar tag vocabulary tend to tag alike, which indicates a preference for personalized tag recommendation services.

It is also notorious the reasonable good results achieved by the *most global frequent tags* recommender, which indicates its adequacy for cold-start related problems, where just a few tags are available in the system.

In future work we plan to reproduce the same experiments with different datasets from different domains to confirm the results here presented. We also want to refine the CF algorithms exploring different combinations between the user similarities obtained from the two profile matrices, i.e., user-resources and user-tags. Moreover,

[5] T-test for a significance level of 0.05.

we will compare the CF approach with more complex models such as multi-label and relational classifiers.

7 Acknowledgments

This work is supported by CNPq, an institution of Brazilian Government for scientific and technologic development.

References

BENZ, D., TSO, K., SCHMIDT-THIEME, L. (2006): Automatic Bookmark Classification: A Collaborative Approach. In: Proceedings of the Second Workshop on Innovations in Web Infrastructure (IWI 2006), Edinburgh, Scotland.

BERNERS-LEE, T., HENDLER, J. and LASSILA, O. (2001): "Semantic Web", Scientific American, May 2001.

BROOKS, C. H., MONTANEZ, N. (2006): Improved annotation of the blogosphere via autotagging and hierarchical clustering. New York, NY, USA : ACM Press, WWW '06: Proceedings of the 15th international conference on World Wide Web : 625Ů632.

DESHPANDE, M. and KARYPIS, G. (2004): Item-based top-n recommendation algorithms. ACM Transactions on Information Systems, 22(1):1-34.

GOLBER, S., HUBERMAN, B.A. (2005): "The Structure of Collaborative Tagging System", Information Dynamics Lab: HP Labs, Palo Alto, USA, available at: http://arxiv.org/abs/cs.DL/0508082

HEYMANN, P. and GARCIA-MOLINAY, H. (2006): Collaborative Creation of Communal Hierarchical Taxonomies in Social Tagging Systems. Technical Report InfoLab 2006-10, Department of Computer Science, Stanford University, Stanford, CA, USA, April 2006.

HOTHO, A., JAESCHKE, R., SCHMITZ, C., STUMME, G. (2006): Information Retrieval in Folksonomies: Search and Ranking. Heidelberg : Springer , The Semantic Web: Research and Applications 4011 : 411-426.

MIKA, P. (2005): Ontologies Are Us: A Unified Model of Social Networks and Semantics. In: Y. Gil, E. Motta, V. R. Benjamins and M. A. Musen (Eds.), ISWC 2005, vol. 3729 of LNCS, pp. 522Ű536. Springer-Verlag, Berlin Heidelberg.

MISHNE, G. (2006): AutoTag: a collaborative approach to automated tag assignment for weblog posts. New York, NY, USA : ACM Press , WWW '06: Proceedings of the 15th international conference on World Wide Web : 953Ů954.

SARWAR, B., KARYPIS, G., KONSTAN, J. and REIDL, J. (2001): Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on World Wide Web. New York, NY, USA: ACM Press, pp. 285-295.

XU, Z., FU, Y., MAO J., SU, D. (2006): Towards the Semantic Web: Collaborative Tag Suggestions. Edinburgh, Scotland: Proceedings of the Collaborative Web Tagging Workshop at the WWW 2006.