2009 Ninth International Conference on Intelligent Systems Design and Applications

Using Tag Co-occurrence for Recommendation

Christian Wartena, Rogier Brussee, Martin Wibbels
Novay
Enschede, The Netherlands
firstname.lastname@novay.nl

***Abstract*—Tagging with free form tags is becoming an increasingly important indexing mechanism. However, free form tags have characteristics that require special treatment when used for searching or recommendation because they show much more variation than controlled keywords. In this paper we present a method that puts this large variation to good use. We introduce second order co-occurrence and a related distance measure measure for tag similarities that is robust against the variation in tags. From this distance measure it is straightforward to derive methods to analyze user interest and compute recommendations. We evaluate the use of tag based recommendation on the Movielens dataset and a dataset of tagged books.**

***Keywords*-Recommendation; tags; co-occurrence; distance measure**

I. Introduction

Tagging is becoming an increasingly important tool to help people organize their information in huge item collections. For example, it allows people to bookmark the items they are interested in and to organize them into various topic sets by adding tags. Once sufficiently many items are tagged, the tags can also be used to search for items on a specific topic. Since tags are associated to both items and users, tags can also be used for generating personalized recommendations. However, unlike keywords or subject headings assigned by information professionals, usually tags lack any form of explicit organization and normalization. Therefore, search and recommendation need to be adapted to the characteristics of tagging systems.

In this paper, we present a theoretical basis for analyzing tags. This analysis is used for a method to determine user interests and generate recommendations for new items from these interests.

The organization of this paper is as follows. In section II we discuss related work. In section III and IV we present the statistical techniques used to relate tags, users and items. Our approach to deriving user interests based on clustering tags is discussed in this section as well. In section V we evaluate tag based recommendation.

II. Related work

Automatic content recommendation has already become a mature field of academic study. We refer to [1] for an overview. A number of standard algorithms has evolved. Most of which are based on implicit or explicit feedback from users on items, usually in the form of item ratings. As large collaboratively tagged data collections have only recently become available, there are no standard techniques for tag based recommendation yet.

Two methods to use tags for recommendations have been found in literature. First, tag-aware recommender systems are based on user feedback, but use tags to compute additional user-user or item-item similarities in order to improve the results of collaborative filtering techniques. A representative example of this category is [2].

In the second approach, which we will follow in this paper, recommendations are completely based on tags. Hung et al. [3] base recommendation on the similarity between the set of tags used by a user and the tags of an item. Given an item and a user, they determine for each item tag the most similar user tag. The similarity between the set of item tags and the set of user tags is the sum of all these maximal similarities. As a base for tag similarity, they take co-occurrence of tags in the user-tag matrix.

Firan et al. [4] discuss several variants. In one group of methods, they use collaborative filtering on the user-tag matrix to find a new (larger) set of tags for a user. This set is compared with the tags from an item to compute a user-item similarity. Again, the most similar items are recommended. A similarity measure is defined by the cosine between the tag occurrence vectors. The second group of methods is similar, but directly use the tags of a user, skipping the recommendation of tags. They evaluate their algorithms in a user experiment with last.fm data. The first group of algorithms performs significantly worse than baseline collaborative filtering based on ratings. The second group of algorithms outperforms collaborative filtering. In Jaschke et al. [5], folk rank [6] is used in combination with collaborative filtering.

III. Tag distributions

One of the main characteristics of collaborative tagging systems is that users are not restricted in their choice of tags. Most users experience this as a feature that makes it easy to use the system. At the same time, this freedom is the main bottleneck for using tags in retrieval tasks because different tags might have been used for the same concept, which makes it difficult to find all items relevant for a certain

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.130

tag. To overcome this problem, tags have to be found that are conceptually related.

A natural approach to determine tag similarity is to use co-occurrence patterns of tags. The underlying idea here is that tags that have often been associated to the same items are likely to be semantically related, the so called distributional hypothesis [7], [8] (see also [9], [10] for experimental evidence in the tagging domain). In [10] and [11], we have presented an approach to co-occurrence using second order co-occurrence of tags and shown that it gives better results for tag similarity than direct co-occurrence. Here we only sketch the basic ideas and define the most important concepts.

Whereas first order co-occurrence only looks at the co-occurrence of one tag with one other tag, second order co-occurrence considers the co-occurrence of one tag with all other tags. This whole pattern of co-occurrences is more stable and more informative than single co-occurrence.

Technically, the above means that for each tag one counts all tag co-occurrences, i.e. the number of times each tag is given to an item carying that particular tag. Normalizing by the total number of co-occurring tags, this gives a distribution over tags, which is called the co-occurrence distribution. The intuitive notion of semantic similarity of tags can now be operationalized as the similarity between the co-occurrence distributions. Note, that two tags can have similar co-occurrence distributions while their mutual co-occurrence is actually very low. In [12], the observation has already been made that this is in fact typical for synonyms in texts. Our results indicate that this observation holds for tags as well.

A. Formal setup

For the following we consider a collection of tagged items (or documents) $\mathcal{C} = \{d_1, \dots d_M\}$. Each tag occurrence is an instance of a tag t in $\mathcal{T} = \{t_1, \dots t_m\}$. Let $n(d,t)$ be the number of occurrences of tag t on d, $n(t) = \sum_d n(d,t)$ be the number of occurrences of tag t, and $N(d) = \sum_t n(d,t)$ the number of tag occurrences in d. Now define

$$\begin{aligned} q(t|d) &= n(d,t)/N(d) \text{ the tag distribution of item } d\ , \\ Q(d|z) &= n(d,z)/n(z) \text{ the item distribution of tag } z\ . \end{aligned}$$

The probability distributions $q(t|d)$ and $Q(d|z)$ on respectively the set of tags $\mathcal{T}$ and the corpus $\mathcal{C}$ describe how tag occurrences of a given item d are distributed over different tags, respectively how the occurrences of a given tag z is distributed over different items. Now define the co-occurrence distribution of a tag z as:

$$\bar{p}_z(t) = \sum_d q(t|d)Q(d|z).$$

The co-occurrence distribution is the weighted average of the tag distributions of items, where the weight is the relevance of d for z given by the probability $Q(d|z)$.

We define the similarity between tag distributions as the information theoretic Jensen Shannon divergence of the distributions. We refer to [13] for a definition.

B. Searching with tags

The divergence of co-occurrence distributions allows us to compute similarities between tags and items or between tags and users since each of them can now be represented by a distribution over tags: the co-occurrence distribution, the distribution of assigned tags and the distribution of used tags, respectively.

Using the co-occurrence distribution and the divergence solves a number of problems for searching in tagged collections. To find an item relevant for a (queried) tag, not all items labeled with that tag are naively searched for, but items are retrieved with a tag distribution with small divergence to the co-occurrence distribution of the queried tag. Thus, in some sense, a kind of query expansion is performed in which every term gets a weight.

Consider the following example. Someone searches for the tag *British history*. With naive search he will miss items that are highly relevant but tagged with e.g. *English history*, *history of Great Britain*, *Mediaeval England*, etc. On the other hand, he will find items in a high ranked position that have the tag *British history*, but that only touch upon the subject and are mostly about something else. Using our approach he will find items tagged with many tags related to British history, probably but not necessarily including that tag itself.

Using this representation there is also a natural way to search for items relevant to a whole set of tags: one can simply take the (weighted) average of all co-occurrence distributions, and use that distribution as query. This way of searching forms the core of our recommendation strategy.

IV. User interests

Users assign tags to items. This tells us something about these items but also about those users. Tags that someone frequently uses are likely to reflect some of his interests relevant for the collection under consideration. These interests could be used to recommend new items to the user. Therefore, we need a way to distill the user's interests from the set of tags that he has used. The individual tags might be too detailed to represent interests, since people tend to use dozens to hundreds of different tags. The overall weighted average of all tags on the other hand is expressionless as it blurs all topics to one uniform gray mixture. Thus clustering seems a natural way to go. The importance of clustering tags for browsing tags was also stressed by Begelman et al. [14].

A. Tag clusters

For clustering we use a straightforward agglomerative hierarchical clustering algorithm. Initially each tag is a cluster. Subsequently, in each step two clusters are merged until a stopping criterion is met. To select the clusters that are merged in each step, we determine the pair of clusters for which the sum of the squares of the distances between all their elements is minimal. This criterion guarantees that at each step the option is chosen that yields the best (highest) Calinksi Harabasz index [15].

As a stopping criterion, we require the number of clusters to be equal to the square root of the number of tags, in order to make the complexity within a cluster comparable to the complexity of the set of clusters. Consequently, in our user interface the size of the cloud of words representing the clusters is similar to the clouds of words within the clusters. We also experimented with more data driven stopping criteria, like a maximum Calinski Harabasz index or a maximum Dunn Index. In most cases, such criteria yield clusterings with very many clusters of one element or with only one cluster. This is in line with the findings of [16] for the usage of the Dunn Index as a stopping criterion for document clustering.

We apply the clustering to the set of tags of each user, i.e. there is no overall clustering of tags, topic clusters are determined individually. E.g. the tag *British history* for one user might end up in a cluster about England for one user, while for another user it belongs to a cluster on general history or to a cluster on the history of London. While tags involved in clustering are only those used by a given user, for each tag we use the co-occurrence distribution that is obtained by taking into account the whole collection because the set of items tagged by one user generally is too small to obtain reliable statistics from.

B. Recommendation

Our approach to recommendation is a content-based approach in the sense of [17]. In their terminology our algorithm can be described as a nearest neighbor method, i.e. for a given user we recommend those items that are closest to his profile. As a user profile (the data to characterize a user) we use the (weighted) average co-occurrence distribution of all the tags the user has utilized. As a distance measure we again use the Jensen-Shannon divergence between co-occurrence distributions.

In order to get a better diversity in the results we also implemented a second variant. In this variant the tags of the user are clustered (see section IV-A). We then compute the nearest neighbors for each cluster center. The results from each cluster are combined to produce a final recommendation. In our user interface the user also has the possibility to view the recommendations per cluster. Thus we obtain recommendations with a nice variation of topics like in the approach of [18]. In contrast to their approach the topic diversification is not obtained by re-ranking the results of a single recommendation, but by merging results of recommendations for different detected topics.

V. Evaluation

To test our approach, we have evaluated the effectiveness of our recommendation algorithm using a dataset from LibraryThing and the MovieLens dataset. The results were compared with two rating based recommendation algorithms, item average and matrix factorization, and with recommendation of the most viewed items. The item average algorithm simply suggests the items with the best overall ratings. This algorithm can be seen as a base line. Matrix factorization is known as one of the strongest techniques available. For a description of the matrix factorization algorithm we used, we refer to [19]. The most-viewed algorithm recommends the items that have been viewed by most users.

We considered two variants of the tag based recommendation presented above. The first variant clusters the tags, computes recommendations for each tag cluster and then merges the results. The number of results from each cluster in the composed list is proportional to the number of assignments of tags form each cluster. The second variant does not use clustering.

A. Dataset

We have tested our recommendation algorithm with two datasets: the tagged MovieLens data and a dataset from LibraryThing. LibraryThing is a web service for managing book collections allowing, among other things, to tag books. We used a dataset form LibraryThing [20], [21], collected by Maarten Clements, in which each user has supplied tags and ratings to at least 20 books and each book has received at least 5 tags. This results in a set with characteristics as given in Table I. Up to now, we have only used this dataset for a subjective evaluation. General impression from people we showed the dataset was that our application was indeed suggesting reasonable books when clicking on tags that people considered interesting.

The second set we used is the tagged MovieLens dataset [22]. This dataset is widely used for recommendation research and contains data of users, movie ratings from users, and for a relatively small subset, tags that have been added to these movies. In order to obtain a subset that is suited to compare tag based recommendation with rating based techniques, we only considered the data that were entered after the moment that the first tag was added to the system (2005/12/23 4:49:47). To split up this set in a training and an evaluation set, we assume that a user has viewed a movie if he has either tagged or rated it. We define a view as the relation of a user and a movie that he has viewed. We then split up the dataset by determining the time point such that 80% of the views lies before that point (2008/6/30 12:40:13). The views before that point are used for training, while the views after that point are used for evaluation. This results in a training set with characteristics as given in Table I.

We only compute and evaluate recommendations for users that have assigned at least 1 tag and 1 rating in the training set, and have at least 1 view in the test set in the sense above. This results in a set of 734 users. However, to compute co-occurrence distributions and recommendations we will take into account the whole training set, to make optimal use of available co-occurrence statistics.

	Libr.Thing	ML Train	ML Test
Users	7.279	71.567	5.122
Items	37.232	10.681	9.241
Rated items	37.232	9.779	9.182
Tagged items	37.232	7.040	3.387
Ratings	749.401	1.817.966	457.604
Tags assignm.	2.056.487	83.132	12.442
Unique tags	10.559	14.625	4.440

Table I
CHARACTERISTICS OF USED DATA SET SELECTIONS

In the resulting training set, there are many users that have only tagged one item with only one singe tag. We cannot expect good tag based recommendations for these users. Rather than further reducing our training set we have divided all users in groups of 25 with an ascending number of tag assignments. Results will be presented for each of these groups. Unsurprisingly the number of tag assignments correlates with the number of ratings and the number of views in the test set as shown in Figure 1. Furthermore, we have to note that not all items in the training set are both tagged and rated. Thus the tag based and the rating based algorithms have a slightly different set of items to choose from for recommendation.

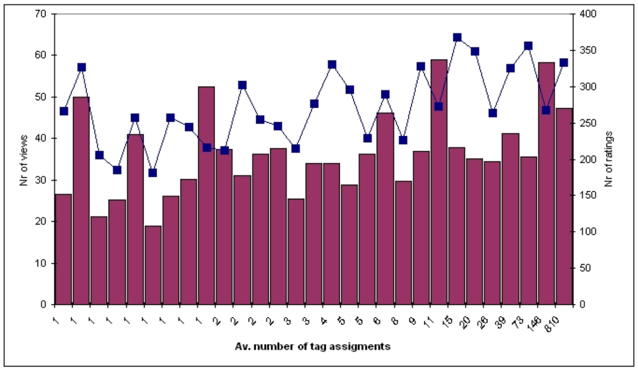

Figure 1. Number of ratings and test views for users with ascending number of tag assignments.

B. Evaluation procedure

Our evaluation criterion consisted of predicting for each user and each algorithm a top 100 list of suggestions of movies not yet viewed according to the training set. We then check which of these items have actually been viewed (i.e. rated or tagged) according to the test set, not taking the eventual rating into account.

The simplest kind of evaluation possible is computing the hit rate, as is done e.g. by Deshpande and Karypis [23]. The hit rate is comparable to the precision in our case. The precision is the number of correctly predicted items divided by the number of all predicted items, which in our case is always 100. However, precision implies that we consider all recommended items that are not in the test set as false positives, while we actually do not know anything about them. We can also compute the recall by dividing the number of hits by the number of items the user has seen in the test set. This implies that we consider all items that are seen but not recommended as false negatives, which is more reasonable.

More advanced methods would include the position of the predicted items on the list, as in the average reciprocal hit rate [23] or also include the rating of the items in the test set, e.g. as in the scoring used by Breese et al. [24]. We cannot use that evaluation metric here since we do not have ratings for all items in the test set

C. Results

The tag based recommendation using the clustering gives results that have an intuitive appeal. The clustering clearly takes into consideration the different interests of a user and produces a more varied and serendipitous list of items. However, the results according to precision and recall as sketched above are slightly worse than those obtained by the variant without clustering. This is in line with results on topic diversification described in [18]. For the kind of evaluation used here it is better to produce as many predictions on the most prominent topic in order to get the largest possibility to hit an item in the test set. In the following, we present numbers only for the variant without clustering.

The hit rate of the test algorithms is given in Table II. It is important to notice that the average number of views of evaluated users in the test set is 36, which is thus also the upper bound for the hit rate. The recall is given in the same table.

Algorithm	Hit rate	Recall
Most viewed	4.00	0.103
Item average	0.84	0.042
Matrix factorization	1.67	0.053
Tag based	1.18	0.029

Table II
HIT RATE AND RECALL FOR 4 ALGORITHMS

As expected, the results of the tag based algorithm are dependent on the number of items the users have tagged. The hit rate and recall for user with ascending number of tag assignments averaged over groups of 25 users is given in Figure 2 and 3.

According to the evaluation metric used, the *most viewed* strategy clearly gives the best results. However, these recommendations are not personalized and probably not very interesting. In fact this indicates the weakness of the evaluation metric, a fact also noticed by [25]. The matrix factorization clearly gives the best results from the remaining algorithms. This is as expected since the algorithm is known as one of the best ones available. Moreover, there are many more ratings than tags. For a large part of the users, the tag based algorithm has to generate a recommendation from one singe tag. If we look at the results for users with 9 or more tags, both precision and recall are competitive with the results from matrix factorization, which can still use a lot more ratings per user. In our opinion this is a remarkable result. However, we should keep in mind that the matrix factorization only predicts items that the user is expected to rate over average, whereas our algorithm does not take that into account. Nevertheless, our tag based algorithm seems to be an interesting technique in situations in which there are tags but no ratings are available.

In the present work we have tacitly assumed that all tags are related to topics. In fact Bischoff et al. [26] found that depending on the source of the tags only about half of tags can be considered topic related. For the music related site they studied, roughly half the tags can be related to genre of the music. Possibly, results can be improved if we take that kind of information into account. This is especially suggested by the results of the cluster based variant. For some users the algorithm generates a cluster with tags like *bad*, *never want to see again* etc. The recommendations for these clusters are currently also merged into the overall list of recommendations.

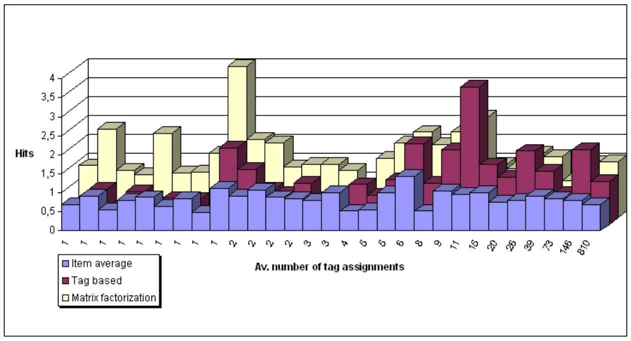

Figure 2. Hit rate for 3 algorithms for groups of 25 users with ascending number of tag assignments.

VI. Conclusion

In this paper, we have investigated the effectiveness of a co-occurrence based distance measure of tags for personalized recommendations. The recommendation is based on the fact that distances can be computed between items, users and tags in a uniform way. To evaluate our approach, we have computed recommendations for users from two datasets that contain both tags and ratings. This enables us to select a subset on which tag based recommendation can be compared with well established rating based recommendation algorithms.

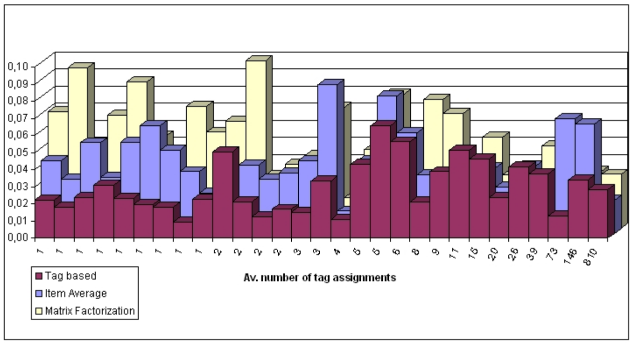

Figure 3. Recall for 3 algorithms for groups of 25 users with ascending number of tag assignments.

As expected the results are not as good as the recommendations obtained by matrix factorization, which has far more data to work on. Surprisingly, the recommendations for users with about 10 or more tag assignments are comparable with matrix factorization. Thus the algorithm seems at least to be an interesting alternative for situations in which no ratings are available.

Future work has mainly to deal with better evaluation metrics. The metrics used in the paper did not take into account the position on the top-n list nor the rating of the items in the test set. These criteria will favor the rating based items. On the other hand, the tag based algorithms, especially the one with clustering, will give a better spread over different topics. Finally, we want to investigate possibilities to use both tags and ratings in one algorithm.

Acknowledgments

The research leading to these results is part of the MyMedia project (http://www.mymediaproject.org) and has received funding from the European Community's Seventh Framework Program (FP7/2007-2011) under grant agreement n 215006. We would like to thank the team from Lars Schmidt-Thieme (Hildesheim University) for making available the matrix factorization software, Maarten Clements (Technical University Delft) for making available his set of LibraryThing data and Mark van Setten for many helpful comments.

References

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, 2005.

[2] K. H. L. Tso-Sutter, L. Balby Marinho, and L. Schmidt-Thieme, “Tag-aware recommender systems by fusion of collaborative filtering algorithms,” in *SAC*, R. L. Wainwright and H. Haddad, Eds. ACM, 2008, pp. 1995–1999.

[3] C. Hung, Y. Huang, J. Hsu, and D. Wu, “Tag-Based User Profiling for Social Media Recommendation,” in *Workshop on Intelligent Techniques for Web Personalization & Recommender Systems at AAAI2008, Chicago, Illinois*, 2008.

[4] C. S. Firan, W. Nejdl, and R. Paiu, “The benefit of using tag-based profiles,” in *LA-WEB*, V. A. F. Almeida and R. A. Baeza-Yates, Eds. IEEE Computer Society, 2007, pp. 32–41.

[5] R. Jäschke, L. B. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme, “Tag recommendations in folksonomies,” in *PKDD*, ser. Lecture Notes in Computer Science, J. N. Kok, J. Koronacki, R. López de Mántaras, S. Matwin, D. Mladenic, and A. Skowron, Eds., vol. 4702. Springer, 2007, pp. 506–514.

[6] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme, “Information retrieval in folksonomies: Search and ranking,” in *ESWC*, ser. Lecture Notes in Computer Science, Y. Sure and J. Domingue, Eds., vol. 4011. Springer, 2006, pp. 411–426.

[7] J. Firth, “A synopsis of linguistic theory 1930-1955,” *Studies in linguistic analysis*, pp. 1–32, 1957.

[8] Z. Harris, *Mathematical structures of language*. Wiley, 1968.

[9] C. Cattuto, D. Benz, A. Hotho, and G. Stumme, “Semantic grounding of tag relatedness in social bookmarking systems,” in *International Semantic Web Conference*, ser. Lecture Notes in Computer Science, A. P. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. W. Finin, and K. Thirunarayan, Eds., vol. 5318. Springer, 2008, pp. 615–631.

[10] C. Wartena and R. Brussee, “Instance-based mapping between thesauri and folksonomies,” in *International Semantic Web Conference*, ser. Lecture Notes in Computer Science, A. P. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. W. Finin, and K. Thirunarayan, Eds., vol. 5318. Springer, 2008, pp. 356–370.

[11] ——, “Topic detection by clustering keywords,” in *DEXA Workshops*. IEEE Computer Society, 2008, pp. 54–58.

[12] H. Schütze and J. Pederson, “A cooccurrence-based thesaurus and two applications to information retrieval,” in *Proceedings of RIA Conference*, 1994, pp. 266–274.

[13] T. Cover and J. Thomas, *Elements of information theory*. Wiley-Interscience, 2006.

[14] G. Begelman, P. Keller, and F. Smadja, “Automated Tag Clustering: Improving search and exploration in the tag space,” in *Collaborative Web Tagging Workshop at WWW2006, Edinburgh, Scotland*, 2006.

[15] T. Caliński and J. Harabasz, “A dendrite method for cluster analysis,” *Communications in Statistics-Simulation and Computation*, vol. 3, no. 1, pp. 1–27, 1974.

[16] S. Meyer zu Eissen and B. Stein, “Analysis of clustering algorithms for web-based search,” in *PAKM*, ser. Lecture Notes in Computer Science, D. Karagiannis and U. Reimer, Eds., vol. 2569. Springer, 2002, pp. 168–178.

[17] M. J. Pazzani and D. Billsus, “Content-based recommendation systems,” in *THE ADAPTIVE WEB: METHODS AND STRATEGIES OF WEB PERSONALIZATION. VOLUME 4321 OF LECTURE NOTES IN COMPUTER SCIENCE*. Springer-Verlag, 2007, pp. 325–341.

[18] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, “Improving recommendation lists through topic diversification,” in *WWW*, A. Ellis and T. Hagino, Eds. ACM, 2005, pp. 22–32.

[19] S. Rendle and L. Schmidt-Thieme, “Online-updating regularized kernel matrix factorization models for large-scale recommender systems,” in *RecSys*, P. Pu, D. G. Bridge, B. Mobasher, and F. Ricci, Eds. ACM, 2008, pp. 251–258.

[20] http://ict.ewi.tudelft.nl/ maarten/LT.

[21] M. Clements, A. P. de Vries, and M. J. T. Reinders, “Detecting synonyms in social tagging systems to improve content retrieval,” in *SIGIR*, S.-H. Myaeng, D. W. Oard, F. Sebastiani, T.-S. Chua, and M.-K. Leong, Eds. ACM, 2008, pp. 739–740.

[22] http://www.grouplens.org/taxonomy/term/14.

[23] M. Deshpande and G. Karypis, “Item-based top- recommendation algorithms,” *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 143–177, 2004.

[24] J. S. Breese, D. Heckerman, and C. M. Kadie, “Empirical analysis of predictive algorithms for collaborative filtering,” in *UAI*, G. F. Cooper and S. Moral, Eds. Morgan Kaufmann, 1998, pp. 43–52.

[25] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. Riedl, “Evaluating collaborative filtering recommender systems,” *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, 2004.

[26] K. Bischoff, C. S. Firan, W. Nejdl, and R. Paiu, “Can all tags be used for search?” in *CIKM*, J. G. Shanahan, S. Amer-Yahia, I. Manolescu, Y. Zhang, D. A. Evans, A. Kolcz, K.-S. Choi, and A. Chowdhury, Eds. ACM, 2008, pp. 193–202.

[27] A. P. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. W. Finin, and K. Thirunarayan, Eds., *The Semantic Web - ISWC 2008, 7th International Semantic Web Conference, ISWC 2008, Karlsruhe, Germany, October 26-30, 2008. Proceedings*, ser. Lecture Notes in Computer Science, vol. 5318. Springer, 2008.