SemRec: A Semantic Enhancement Framework for Tag based Recommendation*

Guandong Xu [1,3] **Yanhui Gu** [2] **Peter Dolog** [3] **Yanchun Zhang** [1] **Masaru Kitsuregawa** [4]
[1]Center for Applied Informatics, Victoria University, Australia
[2]Department of Information and Communication Engineering, University of Tokyo, Japan
[3]Department of Computer Science, Aalborg University, Denmark
[4]Institute of Industrial Science, University of Tokyo, Japan
[1]{Guandong.Xu,Yanchun.Zhang}@vu.edu.au [2,4]{guyanhui,kitsure}@tkl.iis.u-tokyo.ac.jp [3]dolog@cs.aau.dk

Abstract

Collaborative tagging services provided by various social web sites become popular means to mark web resources for different purposes such as categorization, expression of a preference and so on. However, the tags are of syntactic nature, in a free style and do not reflect semantics, resulting in the problems of redundancy, ambiguity and less semantics. Current tag-based recommender systems mainly take the explicit structural information among users, resources and tags into consideration, while neglecting the important implicit semantic relationships hidden in tagging data. In this study, we propose a *Semantic Enhancement Recommendation* strategy (SemRec), based on both structural information and semantic information through a unified fusion model. Extensive experiments conducted on two real datasets demonstrate the effectiveness of our approaches.

Introduction

Nowadays social websites have become a major trend in Web 2.0 environment, enabling abundant social tagging data available. Making use of social tagging data for recommendation is emerging as an active research topic in the field of recommender systems recently. Traditional recommender systems focus on the explicit rating data of users , e.g., movie ratings, to gain the user preference and make predictions for new items. Different from rating data, social tagging data does not contain users explicit preference information on resources, instead, reflecting the personalized perceptions on resources by users. In particular, such data involves three types of objects, i.e., user, resource and tag. These differences bring in new challenges as well as opportunities to deal with recommendation problems in the context of social tagging systems.

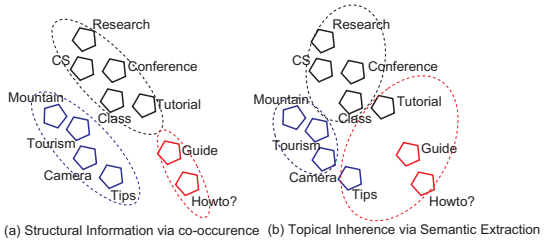
Although we can treat the relationships extracted from the triple-dimensional tagging data, e.g., the correlation between user and resource (annotated or not) as a pseudo-rating, we still have to face several problems: (1) Due to the uncontrolled characteristics of annotations, which results in the severe redundancy and ambiguity of tags, we cannot easily distinguish the topics which the tags present. (2) In annotation services, tags and resources follow the power law distribution, which indicates that the tagging data is very sparse. (3) The annotation data may not properly capture the interests of users because it only reveals the explicit structural information but insufficient implicit semantic relationships. All these problems largely hinder the applicability of the traditional collaborative filtering algorithms in tag-based recommender systems.

There are a number of studies addressing the above difficulties, e.g., in (Shepitsen et al. 2008) clustering was employed to uncover the tag clusters from the co-occurrence of tags annotated on resources. The discovered tag clusters could be considered as one kind of explicit topic information from the structural view of the tagging data, however, little implicit semantic knowledge is seen at this stage. Therefore if we can leverage the additional semantics hidden in the tagging data, we are able to achieve the improved recommendations further. Let's take the following example to illustrate the problem we address. As shown in Figure 1, there are 11 annotated tags. Based on their co-occurrence in annotation data, they might be assigned into three distinct tag groups. However via latent semantic analysis, we can further notice that the tags of "Howto, Guide, Tips and Tutorial" actually form a semantic topic of "Howto" in hidden topic space (Krestel, Fankhauser, and Nejdl 2009). Thus suppose that a user annotates the tag of "tutorial", we then can implicitly identify the user's possible interest on "Howto" (red ellipse) in addition to the topic of "Research" (black ellipse), which can substantially enhance the user's interest capturing, in turn, facilitating recommendations.

Motivated by the above scenario, in this paper we propose a Semantic Enhancement Recommendation approach (SemRec), which combines the strengths of explicit structural and implicit semantic analysis into a unified scheme. The main idea is that by using the proposed approaches, the original pure tag vector expressions could be semantically transformed into two new conceptual and semantic spaces, over which the similarity computations are carried out and fused together. To our best knowledge, there is only a limited number of previous works addressing the semantic enhancement for tag-based recommendation in previous studies.

The main contributions of this paper are as follows:

*This work has been partially supported by EU FP7 ICT project M-Eco: Medical Ecosystem Personalized Event-Based Surveillance under grant No. 247829.



(a) Structural Information via co-occurence (b) Topical Inherence via Semantic Extraction

Figure 1: An Example of Semantic Enhancement in Tagging

- We address the limitations of tags in tag-based recommendation, such as tag ambiguity, redundancy and less semantics via semantic enhancement approaches.
- We propose a semantic enhancement fusion framework for personalized recommendation, which combines the clustering and hidden topic model.
- We conduct experiments on real datasets to evaluate the effectiveness of the proposed approach and investigate the optimization of the most important parameters of our model.

The remainder of the paper is structured as follows: In Section 2, we review the related work. In Section 3, we present the preliminaries of the tagging data model and standard tag-based recommendation. We intensively discuss the proposed approach in Section 4, where clustering, LDA analysis and fusion framework are given. Section 5 reports the experimental setting and evaluation results. And eventually we conclude the paper and outline the future work in Section 6.

Related Work

In this part, we review the literatures related to our work from the following aspects:

Tag-based Personalized Recommendation (Durao and Dolog 2010) developed a multi-factorial tag-based recommender system, which took various lexical and social factors of tags into the similarity calculation. (Shepitsen et al. 2008) proposed a personalized recommendation system by using hierarchical clustering. In this approach, instead of using the pure tag vector expressions, a preprocessing on tag clustering was performed to find out the tag aggregates for personalized recommendation. (Zhang, Zhou, and Zhang 2010) aimed to integrate the diffusion on user-item-tag tripartite graphs to improve the recommendation of state-of-the-art techniques.

Semantic Enhancement in Recommendation In (Krestel, Fankhauser, and Nejdl 2009), LDA is applied for tag recommendation. For a new resource with a few tags, the topic distribution was inferred and the dominant topics were determined. By referring to the representative tags corresponding to these dominant topics, the top tags were finally chosen for tag recommendation. In (Harvey et al. 2010), the authors extended the LDA topic model to include user data and use the estimated probability distributions in order to provide personalized tag suggestions to users. Different from these methods, our proposed approach focuses on leveraging the topic model for personalized recommendation rather than tag suggestion.

Information Fusion in Recommendation (Groh 2007) combined social network data with tagging for neighborhood generation. (Konstas, Stathopoulos, and Jose 2009) adopted Random Walk with Restart to model the social tagging in a music track recommendation system. In addition, (Hummel et al. 2007) proposed an online social recommender system attempting to incorporate more social information for recommendation generation.

Preliminaries

Social Tagging Data Model

In this paper, we work with tagging data. A typical social tagging system has three types of entities, users, tags and resources which are interrelated with one another. Social tagging data can be viewed as a set of triples (Heymann, Ramage, and Garcia-Molina 2008; Guan et al. 2010). Each triple (u, t, r) represents an observation of a user u annotating a tag t on a resource r. A social tagging system can be described as a four-tuple collection - there exist a set of users, U; a set of tags, T; a set of resources, R; and a set of annotations, A^n. We denote the data in the social tagging system as D and define it as: $D =< U, T, R, A^n >$. The annotations, A^n, are represented as a set of triples containing a user, tag and resource defined as: $A^n \subseteq < u, t, r >: u \in U, t \in T, r \in R$.

Standard Tag-based Recommendation

The standard tag-based recommendation is principally similar to a process of traditional information retrieval but with an additional input of the user tagging preference for personalization (or called personalized recommendation). The procedure consists of two steps of search and personalization. The first step produces a list of candidate resources r_s based on the similarity computation between the query tag issued by a user and all resources in terms of term frequency - inverse document frequency (tf-idf).

The second step utilizes the tagging preference of users to make the personalization. Under the vector space model, each user, u, is modeled as a vector (also called user profile) over a set of tags, where $w(t_i)$, in each dimension corresponds to the relationship of a tag t_i with this user, u, $\vec{u} =< w(t_1), w(t_2), ..., w(t_{|T|}) >$. Likewise each resource, r, can be modeled as a vector (i.e., resource profile) over the same set of tags, $\vec{r} =< v(t_1), v(t_2), ..., v(t_{|T|}) >$, where $v(t_i)$ represents the relationship of a tag t_i with this resource. After that, the similarity computation, e.g., cosine measure, of the target user profile u and the candidate resource profiles r_s selected by the first step, is performed, $sim(u, r), r \in r_s$, to further generate the personalized resources based on various recommendation strategies. The distinction of the tag-based recommendation from the standard information search is that here the recommendation is derived from, not only the query itself, but also the user tagging preference (i.e., personalization).

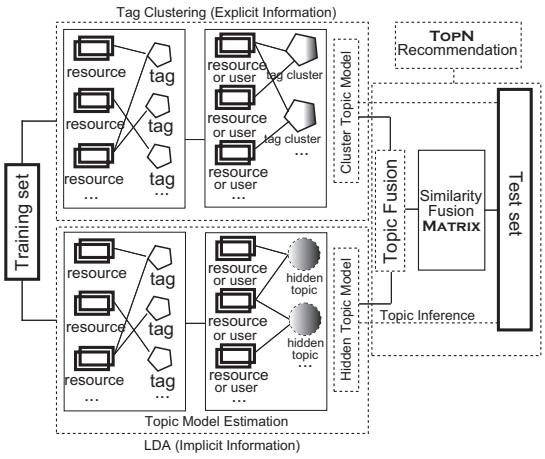


Figure 2: The Framework of SemRec

Semantic Enhancement Recommendation

As discussed above, our major aim is to utilize semantic enhancement approaches to improve recommendations. To fulfill this aim, we propose a novel hybrid framework by combining the latent semantic analysis with the clustering algorithm into a unified fusion scheme. In the following part, we will briefly describe the structure of the framework and the proposed algorithms.

Overview of the Proposed System

The overall framework of our approach shown in Figure 2 consists of two steps. In the semantic enhancement (i.e model training) step, we first conduct the semantic extraction on a training data by using tag clustering and hidden topic estimation approaches, respectively. The tag clustering is to find out the aggregates of tags in tagging activities, while the hidden topic estimation and inference procedure is to capture the topic preference distribution of users and resources at the hidden topic level.

As a consequence, the clustering and LDA manipulations result in two dimensionality-reduced vector spaces in terms of tag clusters and hidden topics. Therefore after these procedures, on the one hand, users and resources are re-expressed by the derived tag cluster space instead of the original pure tag vector space. On the other hand, via LDA estimation and inference, we can also obtain the topic preference distribution of users and resources, which is used to capture the semantic similarity of the users and resources.

In the second personalized recommendation step, we compute the similarity scores between users and resources over the common topics (i.e., tag clusters) and the hidden topics (i.e., via LDA), respectively, and then adopt a fusion mechanism to produce a final score using a tunable parameter λ, which is empirically determined. We eventually select the top-N resources as the recommendations.

Semantic Enhancement Approaches

Tag Clustering The tag clustering is to tackle tag redundancy and to reveal tag aggregates. The user-resource-tag relationships can be represented by a tripartite graph . We decompose the tripartite graph into a bipartite graph along the dimension of resources (Noll and Meinel 2007). Based on the bipartite graph between resources and tags, we can build a resource-tag matrix, in which each row vector of tag over resources denotes whether this tag has been annotated on these resources by various users. The element of the matrix could be represented by a binary or weighting value. Then we employ the Hierarchical Agglomerative Clustering algorithm on this matrix for clustering tags (Shepitsen et al. 2008).

Upon the tag clusters, each user or resource could be re-expressed by a vector over tag clusters. For instance, after performing this procedure of clustering the tags, we obtain two vectors $\vec{u}_{TC} =< w_{TC}(TC_1), w_{TC}(TC_2), \cdots, w_{TC}(TC_{|TC|}) >$ and $\vec{r}_{TC} =< v_{TC}(TC_1), v_{TC}(TC_2), \cdots, v_{TC}(TC_{|TC|}) >$. Furthermore, we utilize the cosine measure to compute the similarity between $\vec{u}$ and $\vec{r}$ at the granularity level of tag aggregations.

$$CosSim_{TC}(\overrightarrow{u}, \overrightarrow{r}) = \frac{\overrightarrow{u}_{TC} \cdot \overrightarrow{r}_{TC}}{\left\| \overrightarrow{u}_{TC} \right\| \cdot \left\| \overrightarrow{r}_{TC} \right\|} \quad (1)$$

Semantic Extraction via LDA Model Theoretically, LDA is a probabilistic generative model for a text corpus. The basic idea of LDA is based on the hypothesis that a person has certain topics in mind when writing an article. To address a topic, the author needs to pick up a word with a certain probability from a bag of words reflecting that topic. In this manner a resource is represented as random mixtures over latent topics and each topic is characterized by a set of related words with a probability distribution. As such, the intuition behind LDA is to uncover this latent topic structure via estimating the probability distribution of the original co-occurrence activities and to capture the correlations between the implicit topics and their representative words in a probabilistic space. Especially in the context of social tagging systems where different users are annotating resources, the obtaining topics represent the commonly shared perceptions of the resources by collaborative users, and the tags of the specific topic constitute a common vocabulary contributed to the topic.

In LDA generative model, a resource $d_m = \{w_{m,n}, n = 1, \cdots, N_m\}$ is generated by picking a distribution over the topics from a Dirichlet distribution. And given the topic distribution, we pick the topic assignment of each specific word. Then the topic assignment for each word $w_{m,n}$ is calculated by sampling a particular topic from the multinomial distribution of $z_{m,n}$. Thus given Dirichlet parameters α and β, we can formulate a joint distribution of a resource d_m, a topic mixture of d_m, i.e., θ_m , and a set of N_m topics, i.e., z_m as follows.

$$P_r(\theta_m, z_m, d_m|\alpha, \beta) = P_r(\theta_m|\alpha) \prod_{n=1}^{N_m} P_r(w_{m,n}|z_{m,n}) P_r(z_{m,n}|\theta_m)$$

And integrating over θ_m, $z_{m,n}$ and summing over z_m , we obtain the likelihood of the resource d_m:

$$P_r(d_m|\alpha, \beta) = \int P_r(\theta_m|\alpha) \prod_{n=1}^{N_m} P_r(w_{m,n}|z_{m,n}) P_r(z_{m,n}|\theta_m) d\theta_m \quad (2)$$

In general, estimating the Dirichlet parameters of LDA is performed by maximizing the likelihood of the whole

resources. In particular, given a corpus of resources $D = \{d_m, m = 1, ..., M\}$, we aim to estimate the parameters of α and β that maximize the log likelihood of the data:

$$(\alpha_{est},\beta_{est})=\arg\max_{\alpha,\beta} \ell(\alpha,\beta)=\max \sum_{m=1}^{M} \log P_r(d_m|\alpha,\beta) \quad (3)$$

However the direct computing for the parameters α and β is intractable due to the nature of the computation. Here we employ the variational EM algorithm (Blei, Ng, and Jordan 2003) to estimate the variational parameters that maximize the total likelihood of the corpus with respect to the model parameters of α and β.

As indicated in the previous section, the aim of employing LDA model training is to reveal the triple relationship between resource, term (in this case equivalent to tag) and hidden topic. In calculation, the triad correlations are modeled by two probability matrices, namely the topic distribution matrix of resources and the tag assignment matrix to topics in terms of probability distributions. In addition, when a user is selected or a new resource is entered, represented by a set of expressive tags, its topic preference distribution can also be further inferred by using this learned model. As a result of LDA training and inference, the user and resource vector expressions are accordingly re-parameterized by the vectors over the hidden topics. In this manner, the user and resource vector forms are expressed as follows:

$$\vec{u}_{HT}=\langle w_{HT}(HT_1),w_{HT}(HT_2),\cdots,w_{HT}(HT_{|TP|})\rangle$$

$$\vec{r}_{HT}=\langle v_{HT}(HT_1),v_{HT}(HT_2),\cdots,v_{HT}(HT_{|TP|})\rangle$$

where $w_{HT}(HT_j)$ and $v_{HT}(HT_j)$ denote the derived topic preference weight on topic HT_j in the transformed user and resource vector expression.

In a similar way, we calculate their cosine similarity $CosSim_{HT}(\vec{u}, \vec{v})$.

$$CosSim_{HT}(\overrightarrow{u},\overrightarrow{r})=\frac{\overrightarrow{u}_{HT}\cdot\overrightarrow{r}_{HT}}{\|\overrightarrow{u}_{HT}\|\cdot\|\overrightarrow{r}_{HT}\|} \quad (4)$$

Similarity Fusion for Recommendation

After applying the above procedures of tag clustering and hidden topic estimation and inference, we obtain two types of similarity between any user and resource pair, i.e., the cosine similarity on tag clusters and on hidden topics. These similarity scores capture the correlation between user and resource from the perspective of common topics and latent semantic topics of tagging behaviors. Eventually we introduce a score fusion mechanism to accommodate these similarity scores in a unified scheme.

$$Sim(\overrightarrow{u},\overrightarrow{r})=\lambda\cdot CosSim_{HT}(\overrightarrow{u},\overrightarrow{r})+(1-\lambda)\cdot CosSim_{TC}(\overrightarrow{u},\overrightarrow{r}) \quad (5)$$

where λ is a fusion factor, which is used to adjust the weight of similarity score over the hidden topics in the fusion process. The final step is to generate the recommendation based on the fusion scores. For each target user, we compute the similarity scores via eq.(5) and choose the resources with the top-N similarity scores as the recommended resources for the user.

Table 1: Statistics of Experimental Datasets

Property	MedWorm	MovieLens
Number of users	949	4,009
Number of resources	261,501	7,601
Number of tags	13,509	16,529
Total entries	1,571,080	95,580
Average tags per user	132	11
Average tags per resource	5	9

Experimental Evaluations

To evaluate our approach, we conducted extensive experiments. We performed the experiments using an Intel Core 2 Duo CPU (2.0GHz) workstation with a 1G memory, running Red Hat Linux 4.1.2-33. All the algorithms were written in C. We conducted experiments on two real datasets, **MedWorm**[1] and **MovieLens**[2].

The first dataset was crawled from the article repository in MedWorm system during April 2010 and then it was ported into our local experimental environment. The second dataset is MovieLens which is provided by GroupLens[3]. It is a movie rating dataset. The statistical result of these two dataset are listed in Table 1.

Evaluation Methodology and Metrics

We utilized the standard metrics in the area of information retrieval to evaluate our approaches. For each dataset, we randomly divided the whole dataset into two parts by 80% (Training set) and 20% (Test set). Here we use *Precision* and *Hit-Ratio* as evaluation metrics. In precision evaluation, for each given user from the test set, we determine the Top-N resources as recommendation based on the generated similarity fusion score. Then we count the total number of resources which are simultaneously occurred in the recommended resource list and real resources for each user and calculate the ratio of this number to the recommendation size as the precision : $precision = \frac{|\mathrm{Recommendation}\cap GroundTruth|}{|\mathrm{Recommendation}|}$. In Hit-Ratio evaluation, for each tagging data, we used the "leave-one-out" strategy, i.e., using the user and tag as an input for personalized recommendations and leaving the resource as the ground truth. By comparing the Top-N recommendations with the left ground truth of resource, we can determine whether the real resource is within the recommended resource list, i.e., $hit_i = 1$, otherwise $hit_i = 0$. By averaging the sum of hits in the test set, we can calculate the Hit-Ratio: $Hit\text{-}Ratio = \frac{\sum_{i=1}^{N} hit_i}{N}$, where N is the size of test set. From the definitions, we can see that Precision and Hit-Ratio measure the recommendation performance from the overall and top recommendation views.

In order to evaluate our semantically enhanced recommendation approach, we also run comparative experiments. In this paper we choose two existing methods - one is pure tag based recommendation (Noll and Meinel 2007) (denoted as PT) and the other is using Hierarchical Agglomerative

[1] http://www.medworm.com/

[2] http://www.movielens.org/

[3] http://www.grouplens.org/

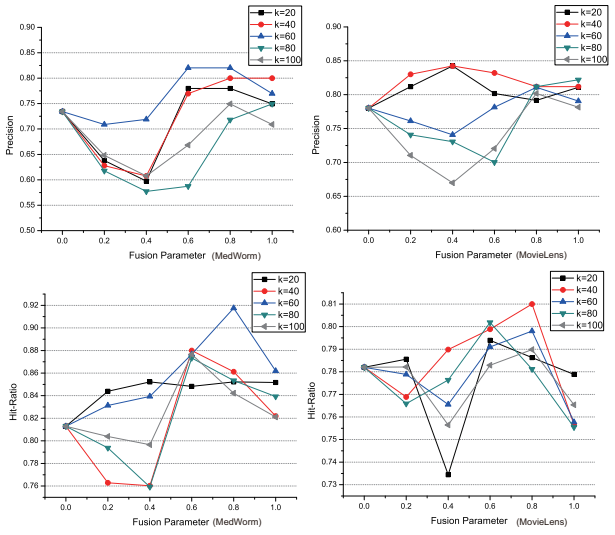


Figure 3: The Impact of Topic Number Selection on Recommendation. (All result @ Top 20)

Clustering (Shepitsen et al. 2008) (denoted as TC) as baselines. For more comparisons, we also conduct recommendations based on the hidden topic model standalone (denoted as HT) and the proposed the semantic fusion approach (denoted as Fusion). We report the experiments and discussions in the following parts.

Experimental Results and Discussions

In this section, we report the experimental results of improved recommendation performance in comparison to two baseline approaches, i.e., using pure tag vector and hierarchical clustering. We also investigate the parameters of hidden topic number K, the clustering coefficient D and the fusion factor λ.

The Impact of Hidden Topic Number As seen, there are a number of technical issues having the impacts on the recommendation performance, e.g., the hidden topic number, the clustering coefficient and the fusion factor. And we empirically observed that the settings of there parameters closely rely on the datasets we are choosing to conduct the experiments. First let us see the determination of hidden topic number. As mentioned above, basically three factors affect the recommendation. In order to accurately assess the impact of changing hidden topic number, we conducted the experiments varying the hidden topic number and fusion factor but fixing the clustering coefficient (in this case the clustering coefficient D= 0.7 and 0.3 for MedWorm and MovieLens, respectively). We select the hidden topic number being K=20, 40, 60, 80 and 100 and train the hidden topic model under these settings. Then for these K settings, we vary the fusion factor from 0 to 1 with a step of 0.2. The results @Top20 are depicted in Figure 3 in terms of precision and Hit-Ratio. We can see from the chart, for MedWorm, the recommendation at K=60 mostly achieves th best results, while for MovieLens, the finding indicates the optimal number is at K=40. The explanation to this observation is probably because that the topics hidden in MedWorm is relative much broader than those in MovieLens, resulting in the selection of a bigger number of topics.

Table 2: The Selection of Cluster Coefficient D

	MedWorm		MovieLens	
D	Precision	Hit-Ratio	Precision	Hit-Ratio
1	0.621	0.732	0.590	0.702
0.9	0.702	0.774	0.709	0.705
0.8	0.793	0.888	0.810	0.710
0.7	**0.831**	**0.918**	0.830	0.722
0.6	0.830	0.863	0.830	0.764
0.5	0.820	0.831	**0.842**	0.731
0.4	0.820	0.810	0.820	0.789
0.3	0.803	0.802	**0.838**	**0.790**
0.2	0.710	0.808	0.799	0.772
0.1	0.656	0.792	0.731	0.764

The Selection of Tag Cluster Coefficient In this experiment, we would like to evaluate the selection of tag cluster number, which is determined by the cluster division coefficient (D) in hierarchical clustering. We choose the selected topic number of 60 and 40, and the fusion factor λ being 0.8 and 04, for MedWorm and MovieLens, respectively. We run the evaluations using D from 0.1 to 1 and the results @Top20 are depicted in Table 2. As we expect, for different datasets, we can conclude different findings. For MedWorm, the precision and Hit-Ratio values are gradually increasing at first and reach a maximum (at D=0.7), and then decreasing when D varies from 0.1 to 1. The changes of precision and Hit-Ratio values of MovieLens are similar to MedWorm, but the climaxes are slightly different. In the case of MovieLens, the best recommendation results of precision achieved are at D=0.3 or 0.5 (two climaxes) whereas the biggest value of Hit-Ratio is occurred at D=0.3. Since the higher value of D results in the more clusters generated, the optimized values of clustering coefficient D imply that we should choose a larger cluster number for MedWorm but a smaller number for MovieLens. These findings are consistent with the experimental setting of hidden topic number. So in later experiments, we empirically set the D being 0.7 and 0.3 for two datasets, respectively.

The Optimization of Fusion Factor Another important parameter is the fusion factor λ, which is to tune the significance weights of tag clusters and hidden topics in the fusion formula. We conduct the experiments with various λ settings from 0.2 to 1 with a step of 0.2. We summarize the results in Table 3 and 4. From the tables, it is clear that $\lambda = 0.8$ is the best setting for MedWorm, whereas for MovieLens $\lambda = 0.4$ should be chosen. The difference in λ for these two datasets suggests that we should give a higher weight to the hidden topic model than the tag cluster model for MedWorm, which means the derived topic model is able to facilitate the recommendation more effectively than tag clusters. However for MovieLens the finding is in opposite. The rational for this conclusion is probably because that topic distribution hidden in MedWorm is of higher quality than that of MovieLens.

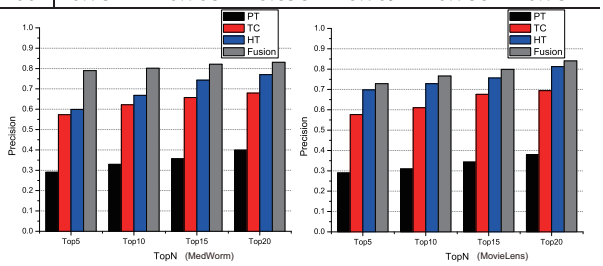
Overall Recommendation Comparisons Upon the parameters empirically optimized by the above steps, we conduct the experiments to compare the overall recommendation performance of our proposed approaches with the base-

Table 3: The Impact of Fusion Factor λ (MedWorm)

K	λ=0.0	λ=0.2	λ=0.4	λ=0.6	λ=0.8	λ=1.0
20	0.679	0.629	0.583	0.743	0.763	0.749
40	0.679	0.639	0.600	0.760	0.795	0.800
60	0.679	0.710	0.721	0.821	**0.831**	0.770
80	0.679	0.608	0.579	0.592	0.707	0.735
100	0.679	0.638	0.610	0.613	0.737	0.694

Table 4: The Impact of Fusion Factor λ (Movielens)

K	λ=0.0	λ=0.2	λ=0.4	λ=0.6	λ=0.8	λ=1.0
20	0.784	0.799	**0.839**	0.799	0.787	0.811
40	0.7845	0.830	**0.838**	0.822	0.813	0.812
60	0.784	0.765	0.740	0.734	0.783	0.791
80	0.784	0.737	0.710	0.697	0.807	0.822
100	0.784	0.708	0.658	0.709	0.786	0.781



(λ=0.7, D=0.7, K=60) (λ=0.4, D=0.3, K=40)

Figure 4: The Overall Precision Comparison Results

line approaches. We calculate the precision @Top-5, 10, 15 and 20 recommendation and the results are shown in Figure 4. From the figure, we can see that the proposed hidden topic approach is able to achieve more satisfactory results than two baseline approaches, for example, with a significant improvement up to 4.4% and 106.4% over the hierarchical clustering and pure tag vector approach respectively for MedWorm. With the proposed hybrid approach the precision values are further increased up to 37.8% and 172.2%. Similar finding can be seen for the MovieLens dataset. As a result, we conclude that the hidden topic approach is able to well deal with the ambiguity, redundancy and less semantic problems of tags for recommendation in tagging systems along with the help of tag clustering approach.

Conclusion

Social tagging systems are becoming a popular information service within the social web era. As an important complementary metadata reflecting user perceptions on web resources, tag based computing is able to facilitate the traditional information processing, such as recommender systems. However the intrinsic drawbacks of tags will result in the difficulties in tag based recommendation. In this paper, we have proposed using the semantic enhancement in tagging systems to improve the recommendation performance. Leveraging the hidden topic distribution derived via LDA is able to capture the correlation between users and resource at the semantic level. Along with the transformed expression in tag cluster space, the fusion of these two expression forms significantly enhances the tag based computing, and in turn, improve the tag based recommendation accordingly. The experiments conducted on two real datasets have demonstrated the superiority of the approaches against the state-of-the-art approaches. The future work may follow the direction of comparisons to other semantic approaches with more tagging datasets.

References

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3(1):993–1022.

Durao, F., and Dolog, P. 2010. Extending a hybrid tag-based recommender system with personalization. In *SAC '10: Proceedings of the 2010 ACM Symposium on Applied Computing*, 1723–1727. New York, NY, USA: ACM.

Groh, G. 2007. Recommendations in taste related domains: Collaborative filtering vs. social filtering. In *In Proc ACM Group07*, 127–136.

Guan, Z.; Wang, C.; Bu, J.; Chen, C.; Yang, K.; Cai, D.; and He, X. 2010. Document recommendation in social tagging services. In Rappa, M.; Jones, P.; Freire, J.; and Chakrabarti, S., eds., *WWW*, 391–400. ACM.

Harvey, M.; Baillie, M.; Ruthven, I.; and Carman, M. 2010. Tripartite hidden topic models for personalised tag suggestion. *Advances in Information Retrieval* 432–443.

Heymann, P.; Ramage, D.; and Garcia-Molina, H. 2008. Social tag prediction. In Myaeng, S.-H.; Oard, D. W.; Sebastiani, F.; Chua, T.-S.; and Leong, M.-K., eds., *SIGIR*, 531–538. ACM.

Hummel, H. G. K.; Berg, B. V. D.; Berlanga, A. J.; Drachsler, H.; Janssen, J.; Nadolski, R.; and Koper, R. 2007. Combining social-based and information-based approaches for personalized recommendation on sequencing learning activities. *Int. J. Learn. Technol.* 3(2):152–168.

Konstas, I.; Stathopoulos, V.; and Jose, J. M. 2009. On social networks and collaborative recommendation. In Allan, J.; Aslam, J. A.; Sanderson, M.; Zhai, C.; and Zobel, J., eds., *SIGIR*, 195–202. ACM.

Krestel, R.; Fankhauser, P.; and Nejdl, W. 2009. Latent dirichlet allocation for tag recommendation. In *Proceedings of the third ACM conference on Recommender systems*, RecSys '09, 61–68. New York, NY, USA: ACM.

Noll, M. G., and Meinel, C. 2007. Web search personalization via social bookmarking and tagging. In *Proceedings of the 6th international The semantic web and 2nd Asian conference on Asian semantic web conference*, ISWC'07/ASWC'07, 367–380. Berlin, Heidelberg: Springer-Verlag.

Shepitsen, A.; Gemmell, J.; Mobasher, B.; and Burke, R. 2008. Personalized recommendation in social tagging systems using hierarchical clustering. In *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, 259–266. New York, NY, USA: ACM.

Zhang, Z.; Zhou, T.; and Zhang, Y. 2010. Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs. *Physica A: Statistical Mechanics and its Applications* 389(1):179–186.