Exploring Generative Models of Tripartite Graphs for Recommendation in Social Media

Charalampos Chelmis
Department of Computer Science
University of Southern California
Los Angeles, CA, USA
chelmis@usc.edu

Viktor K. Prasanna
Ming Hsieh Department of Electrical Engineering
University of Southern California
Los Angeles, CA, USA
prasanna@usc.edu

ABSTRACT

As social media sites grow in popularity, tagging has naturally emerged as a method of searching, categorizing and filtering online information, especially multimedia content. The unrestricted vocabulary users choose from to annotate content however, has often lead to an explosion of the size of space in which search is performed. This paper is concerned with investigating generative models of social annotations, and testing their efficiency with respect to two information consumption oriented tasks. One task considers recommending new tags (similarly new resources) for new, previously unknown users. We use perplexity as a standard measure for estimating the generalization performance of a probabilistic model. The second task is aimed at recommending new users to connect with. In this task, we examine which users' activity is most discriminative in predicting social ties: annotation (i.e. tags), resource usage (i.e. artists), or collective annotation of resources altogether. For the second task, we propose a framework to integrate the modeling of social annotations with network proximity. The proposed approach consists of two steps: (1) discovering salient topics that characterize users, resources and annotations; and (2) enhancing the recommendation power of such models by incorporating social clues from the immediate neighborhood of users. In particular, we propose four classification schemes for social link recommendation, which we evaluate on a real–world dataset from Last.fm. Our results demonstrate significant improvements over traditional approaches.

Categories and Subject Descriptors

G.3 [**Probability and Statistics**]: Probabilistic algorithms; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.3.4 [**Information Storage and Retrieval**]: Systems and Softwares—*Performance evaluation*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MSM '13 May 1, 2013, Paris, France
Copyright 2013 ACM 978-1-4503-2007-8/13/05 ...$15.00.

General Terms

Algorithms, Experimentation, Human Factors, Measurement, Performance

Keywords

Collaborative tagging, graphical models, link recommendation, social media, unsupervised learning, user modeling

1. INTRODUCTION

Users of social media sites involve in rich activities that reveal crucial information about their interests and tastes. Tripartite graphs [8] offer a mechanism to describe and capture users' behaviors and interests in terms of their interaction with online content. For example, mining users' listening frequencies to artists and recording tags with which users annotate artists they are mostly listening to in Last.fm[1], might reveal users' music preferences, but also unveil a hidden structure of annotations and a natural categorization of artists in music genres.

In Last.fm multiple users may listen, bookmark, share or tag an artist. Even though each user performs this task individually, they collectively contribute to the characterization of an artist (resource), resulting in a socially generated set of metadata that describes it. Users annotate resources by choosing tags from an uncontrolled vocabulary according to their style and interests. Resources of the same nature (i.e. topic) may be tagged with different keywords, which may have similar meaning (e.g. synonyms) or with linguistic variations of the same keyword (e.g. "lac" as opposed to "laclippers"). Conversely, the same keyword can be used to annotate resources of different nature due to polysemy. For example, "apple" may be used to describe a story about farmers market or about a new iPhone product.

In this work we explore the use of probabilistic models as a mechanism to address such issues of synonymy, polysemy and tag sparseness and effectively model tripartite graphs in order to simultaneously unveil latent topics, users' interests, and hidden structures of resources and tags in social media. We evaluate three generative models for mining and modeling social media data and assess their recommendation power, reporting our findings. We then show that latent topics, integrated with structural features can be accurate predictors of social ties.

The rest of this paper is organized as follows. Section 2 briefly describes the basic structure of tripartite graphs and

[1] http://www.last.fm

introduces three probabilistic models for tripartite graph generation. Section 3 presents our four scalable social link classification schemes, which combine latent semantics and network proximity. Section 4 demonstrates the effectiveness of our three generative models to capture the hidden structure of social bookmarking sites, as well as evaluates our clustering schemes in the task of social tie recommendation on a real-world dataset. Section 5 summarizes related work and Section 6 concludes with a discussion of the implications of our findings and directions of future work.

2. GENERATIVE MODELS OF COLLABORATIVE ANNOTATIONS IN SOCIAL MEDIA

A tripartite graph is formed by three disjoint node sets: 1) a set of actors (e.g. users) $\mathcal{A} = \{a_1, ..., a_A\}$, 2) a set of concepts (e.g. tags) $\mathcal{C} = \{c_1, ..., c_C\}$ and 3) a set of resources (e.g. artists) $\mathcal{R} = \{r_1, ..., r_R\}$, annotated by actors in $\mathcal{A}$ with concepts from $\mathcal{C}$. More complex hierarchical Bayesian models can be designed by incorporating more types of resources and concepts. The models we describe below can be naturally extended to accommodate other resources and annotation types, such as annotations of Flickr[2] photos, or descriptive text of Youtube[3] videos.

2.1 Modeling Users with Tags

The User-Concept model (**UC**) is an adaptation of the original Latent Dirichlet Allocation (LDA) model [1], where documents are replaced by users' tag collections. LDA treats documents as bags of words. Instead, we model users based on their tag usage, treating tags as vocabulary terms and aggregating annotations users assign to resources. In this model, ϕ denotes the matrix of topic distributions, with a multinomial distribution over N concepts for each of T topics being drawn independently from a symmetric Dirichlet(β) prior. θ is the matrix of user-specific mixture weights for these T topics, being drawn independently from a symmetric Dirichlet(α) prior. For each annotation, z denotes the topic responsible for generating that concept, drawn from the θ distribution for that user, and c is the concept, drawn from the topic distribution ϕ corresponding to z. The generative process is shown in Figure 1a. While this modeling is informative about users' latent interests, it does not provide explicit characterization of resources, nor does it capture the social aspect of tagging, based on which multiple users collectively annotate (similarly use, share, bookmark, etc.) resources. It is thus unclear how the topics used to describe users might be used to also describe resources and unveil their hidden structure, if any.

2.2 Modeling Users with Resources

The User-Resource model (**UR**) is structurally equivalent to LDA [1], however, according to UR, users are modeled based on their interactions with resources. Tags users attach to resources are ignored. In this model, ϕ denotes the matrix of topic distributions, with a multinomial distribution over R resources for each of T topics being drawn independently from a symmetric Dirichlet(β) prior. The matrix of user-specific mixture weights for these T topics, θ, is being drawn independently from a symmetric Dirichlet(α) prior. Each resource r is drawn from the topic distribution ϕ corresponding to z, the topic responsible for generating that resource, drawn from the θ distribution for that user.The underlying graphical model is shown in Figure 1b. Similarly to UC, this model also has limitations. First, UR ignores the social aspect of the tagging process. Further, even though this model captures users' latent interests with respect to resources, it does not provide explicit characterization of tags.

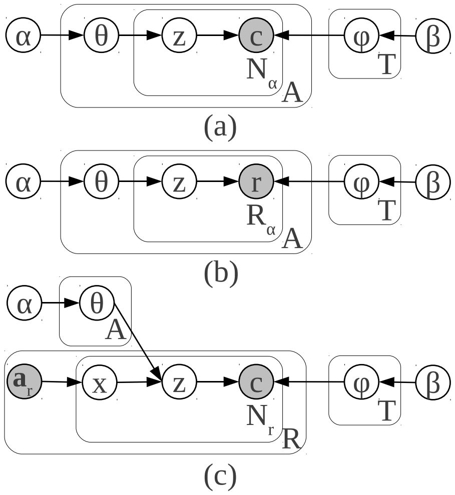

Figure 1: Generative models of collaborative tagging in social media. (a) User-Concept model, (b) User-Resource model, (c) User-Resource-Concept model.

2.3 Joint Modeling of Users, Resources & Concepts

We introduce User-Resource-Concept model (**URT**), a probabilistic model [27], to model user's interests based on resources usage and annotation behavior, and capture the collaboration aspect of tagging. Topics are hidden variables representing categories that naturally split the corpus into clusters of closely related resources. In Last.fm, topics are equivalent to music genres. A group of users $\mathbf{a}_r$, which for the purposes of estimation we assume is observed, collectively annotate (similarly bookmark, share, etc.) resource r with a set of concepts $\mathbf{c}_r$, of size N_r. A collection of R resources is then represented as a concatenation of individual concept vectors $\mathbf{c}$, having $N = \sum_{r=1}^{R} N_r$ concepts in total. For each resource annotation a user x is chosen uniformly at random from $\mathbf{a}_r$. Then, a topic is chosen from a distribution over topics specific to that user, and the annotation is generated from the chosen topic. Each user is associated with a distribution over latent topics θ, chosen from a symmetric Dirichlet (α) prior. Assuming there are T latent topics, the multinomial distribution over topics for each author can be represented as a matrix Θ, of size $T \times A$. Its elements θ_{ta} stand for the probability of assigning topic t to a concept generated by user a. We use θ_a to denote the a^{th} column of the matrix. The mixture weights corresponding to the chosen user are used to select topic z, and a concept is generated according to the distribution ϕ corresponding to that topic, drawn from a symmetric Dirichlet (β) prior. Matrix Φ, of size $C \times T$, denotes the multinomial distributions over tags associated with each topic. ϕ_t represents the probability of

[2] `http://www.flickr.com`

[3] `http://www.youtube.com`

generating concepts from topic t. This generative process is described in graphical form in Figure 1c.

2.4 Parameter Estimation

The number of users, resources and tags in online social media is in the order of millions, hence scalable inferencing is necessary for the applicability of these models in a real world scenario. We adopt collapsed Gibbs sampling [6] to compute the posterior distribution on $\mathbf{z}$ (i.e. the probability of topic mixtures of concepts $P(\mathbf{z} \mid \mathbf{c})$), which captures the hidden structure of topics, and then use the result to infer matrices Φ (i.e. the probability of topics given concepts $P(\Phi \mid \mathbf{c})$) and Θ (i.e. the probability distribution over topics for each user given concepts $P(\Theta \mid \mathbf{c})$). The worst case time complexity of each iteration of the Gibbs sampler is $O(VU_{max}A)$, where A is the number of users, V denotes vocabulary size, and U_{max} is the maximum number of users that can be associated with a resource. As complexity is linear in V, Gibbs sampling can be efficiently carried out on large data sets [27]. Considerable speedup gains can be achieved by optimizing Gibbs sampling and by successfully incorporating recent advances in parallel and cloud computing [20].

3. RECOMMENDATION

All three models that we described above can be utilized to address information needs that are currently overlooked by online social media sites. Novel visualization capabilities may enhance users' experience by enabling social browsing of users, resources and tags with respect to latent topics. Resources mapped to a topic can benefit from tag recommendation to better describe them and improve search. Conversely, tags may be automatically associated to resources that were not originally linked to. Communities might emerge based on users' clustering with respect to common latent interests.

In the rest of this section, we propose four classification schemes that utilize matrix Θ to learn how to recommend appropriate links. Even though we focus on recommending people, our approach can be extended to recommend other "things" (i.e. resources and tags) as well. All classifiers are generated as support vector machines (SVM) with Gaussian radial basis function kernels [4]. We use Sequential Minimal Optimization (SMO) to train our SVM classifiers, requiring memory that grows linearly to the training set size, allowing SMO to handle very large training sets [26]. In testing time, we need to pass a user pair instance onto an SVM model to find the hypothesis (i.e. existence of a link) with the highest confidence. The last classification scheme exploits all previous classifiers, building a hierarchical system.

3.1 Classification Based on Latent Interests

Given user u and her topic distribution $\Theta(:,u)$, we can find similar users v that have a highly similar distribution, hence similar "tastes" to u. The optimization criterion might be based on a similarity metric or a distance measure. For example, the similarity between two users' topic distributions can be measured by applying either (symmetric) Kullback Leibler (KL) divergence, or cosine similarity between the users, by assuming their corresponding topic probabilities as feature vectors [25]. Here, we choose to compute the point–wise squared distance between feature vectors of users u and v. We do this because KL divergence and cosine similarity produce a single feature per users' pair. Instead, using the point–wise squared distance for each pair of users, we provide our classifier a finer grained characterization constituting of T features. The feature vector for a user pair (u,v) is therefore constructed as:

$$F(u,v) = \left[(\Theta(1,u) - \Theta(1,v))^2, \ldots, (\Theta(T,u) - \Theta(T,v))^2\right]. \tag{1}$$

$F(u,v)$ is zero when users u and v are completely aligned with respect to their interests in the latent space, whereas larger values indicate less common interests. The optimization objective is to minimize the distance between users u and v between whom a tie exists. Here we focus solely on similarity of users' interest, ignoring network effects. Considering this scheme we are able to test the hypothesis that social links form on the basis of user homophily or conversely if the social network also plays some role in link formation.

3.2 Models Expansion with Network Structure

Many social link recommendation approaches calculate proximity scores based on graph oriented approaches [22], asserting that the "closer" two users are in the social graph, the more likely they are to become linked in the future. Intuitively, network proximity measures the likelihood of an interaction between two users u and v, regardless of the existence of a path between u and v. Proximity metrics used in prior work include neighborhood based methods and methods based on the ensemble of all paths [22]. In our work, we consider network structure in conjunction to learned latent interests of users. Said et al. [28] examined the impact of social graph structure on users' tastes. Instead, we take a complimentary approach, where we examine the factors that drive social tie creation. Furthermore, we differentiate between local network proximity and similarity that stems from global knowledge of the social graph. This effectively enables us to test if knowledge about the global structure of the network is essential to the recommendation process or if, conversely, local knowledge of the network structure can produce satisfactory recommendations.

3.2.1 *Latent Topics & Local Structure*

For simplicity and computational efficiency, we use the number of common neighbors as feature in this scheme, in order to exploit clues from users' immediate neighborhood. The number of common neighbors between users u and v measures their corresponding neighborhood overlap. It is defined as $CN(u,v) = |\Gamma(u) \bigcap \Gamma(v)|$, where $\Gamma(u)$ is the set of neighbors of user u in the network, and $|\cdot|$ denotes set cardinality. To calculate CN, for each user u and all u's neighbors, we first search all u's neighbors, and then lay out the neighbors of each of u's neighbors, respectively. The time complexity to traverse the neighborhood of a node with k neighbors in a sparse network is $k \ll A$, hence the time complexity for calculating CN is $O(k^2A)$.

To account for user homophily with respect to latent topics, we consider column $\Theta(:,u)$ as a feature vector for user u, and use the standard cosine similarity to compare the feature vectors of two users u and v:

$$\sigma(u,v) = \frac{\sum_t \Theta(t,u)\Theta(t,v)}{\sqrt{\sum_t \Theta(t,u)^2}\sqrt{\sum_t \Theta(t,v)^2}}. \tag{2}$$

This quantity is 0 if u and v share no latent topics, and 1 if they have exactly the same interests. The feature vector for a user pair (u, v) is therefore constructed as:

$$F(u, v) = [\sigma(u, v), CN(u, v)]\,. \tag{3}$$

Aggregation Strategy.

We found that when considering the above feature set, the result is a non separable training sample due to the fact that similarity values between pairs for both positive and negative samples exhibit great variance. This in effect produces very inefficient classifiers, that are either unable to separate the training samples altogether or preform poorly in the recommendation task. To avoid this situation, as well as to reduce the number of training samples provided to the classifier (effectively achieving scalability), we average similarity values over the number of common neighbors. We characterize the average latent similarity of user pairs with k common neighbors in the social network as follows:

$$avg_\sigma(k) = \frac{1}{|p : k_p = k|} \sum_{p:k_p=k} \sigma(p), \tag{4}$$

where p denotes a user pair (u, v) and k_p denotes the number of common neighbors for user pair p. This needs the computation of all user pairs with k common neighbors, for each value of k, and then averaging over all similarity values. We begin by sorting CN by rows and columns in $O(AlogA)$ time (this step can be significantly sped up using better sorting strategies). Searching for user pairs with k common neighbors requires at most $O(A + A) = O(A)$ steps, resulting in $O(K|S_{CN_k}|A)$, where K is the number of unique values of k, and $|S_{CN_k}|$ denotes the maximum cardinality of the set S of user pairs with k common neighbors.

3.2.2 Latent Topics & Global Structure

Instead of using the number of common neighbors, we use shortest distance to capture graph based similarity between users u and v, denoted as $SD(u, v)$. We find SD between every pair of users using Johnson's algorithm [14], resulting in a time complexity of $O(AlogA + AE)$. The feature vector for a user pair (u, v) is therefore constructed as:

$$F(u, v) = [\sigma(u, v), SD(u, v)]\,. \tag{5}$$

Because of the great variance of similarity values, we train this classifier using the average latent similarity of user pairs with shortest distance k in the social network, using Equation (4), with the difference that in this case k_p denotes the shortest distance value for user pair p.

3.3 Ensemble Classification Scheme

We combine the predictions of each of the classifiers we described above, using a consensus mechanism. Each classifier is treated as an expert, providing a vote on whether there is a link between a pair of users or not. We set ensemble weights to have equal values and normalize them such that $\sum_{i=1}^{3} \lambda_{Cl_i} = 1$. The consensus function we use is a weighted binary vote. For a pair of users $p = (u, v)$ and classifier Cl_i we define a prediction function $\xi_{Cl_i}(p)$ such that:

$$\xi_{Cl_i}(p) = \left\{ \begin{array}{ll} 1, & \exists\, e(u, v) \\ 0, & otherwise \end{array} \right. , \tag{6}$$

where $e(u, v)$ denotes a directed edge between users u and v. We compute the consensus score for p as $\sum_{i=1}^{3} \lambda_{Cl_i} \xi_{Cl_i}(p)$. We could have learned different weights for each classifier, indicating our confidence in its predictions. However, this procedure imposes another round of supervised training phase, which would unnecessarily increase the complexity of our approach. In our evaluation section, we show that the majority voting scheme is quite effective in producing high quality recommendations.

Table 1: Dataset description

Number of unique users	1,892
Number of unique artists	17,632
Number of unique tags	11,946
Directed user-user relations	25,434
User-artist relations	92,834
Annotations (user-artist-tag relations)	186,479

4. EXPERIMENTAL ANALYSIS

We conduct several experiments to evaluate the ability of the three generative models (see Section 2) to capture social interactions and uncover the subliminal collective knowledge about online content organization. We further evaluate the performance of our classifiers and compare it to state–of–the–art techniques on a real–world dataset from Last.fm. We performed our experiments on a 2.4 GHz Intel Core 2 Duo, with 2 GB of memory, running Windows 7. All algorithms were implemented in Matlab.

4.1 Dataset

For our experiments we consider $hetrec2011\!-\!lastfm\!-\!2k$, a real–world dataset of 2K users from Last.fm online music system [2] (see Table 1). The dataset includes friend relationships (i.e. user–user) and user–listened to artist relations (i.e. <user,artist,listening count> tuples). The dataset further includes 12K unique tags, used in about 186K annotations (i.e. <user,artist,tag> tuples) of 18K artists. This leads to a vocabulary size of $R = 17,632$ in UR model, and $C = 11,946$ in UC and URC models for this dataset.

4.2 Sample Topics

Figure 2 shows 4 topics (out of 50) learned by the three models. Each topic is illustrated with the top 10 tags (or artists in the case of UR) most likely to be generated conditioned on the topic. Learned topics capture Last.fm's music taxonomy from user-generated annotations. Particularly for the UR model, the top 10 most likely artists in each topic are well-known artists in terms of popularity and fame, and representative samples of the music genres they belong to. Notably, URC topics on the right, match surprisingly well UC rightmost topics. Finally, while most of the topics in our models semantically capture music genres, some topics illustrate some other types of themes discovered. For instance, the left–topmost UC topic captures users preferences in the form of explicitly stated feelings and/or opinions with respect to specific artists.

4.3 Predictive Power

We compare the quality of hidden topics uncovered by the three generative models with respect to perplexity [27]. We divide our dataset into two disjoint sets, such that we retain 90% of the data for training, and the rest for testing. Figure 3 shows the results. URC yields lower perplexity than

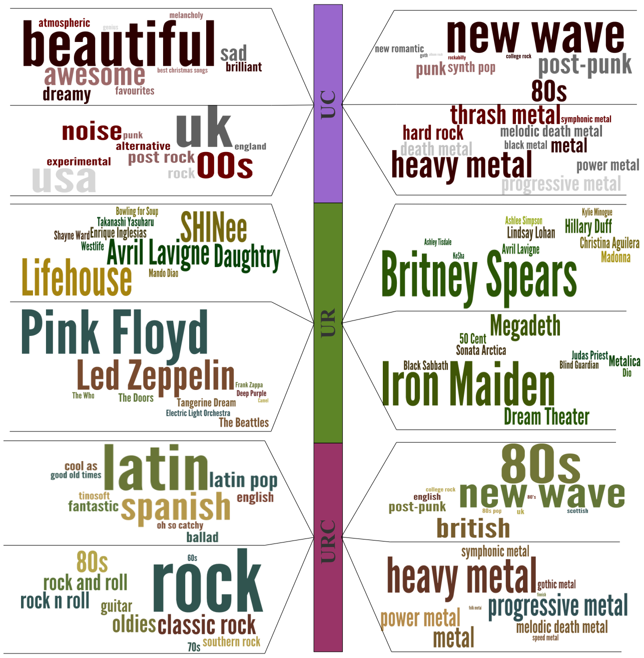

Figure 2: Clouds of top tags and artists for 4 topics (out of 50) learned by the UC, UR and URC models. Size indicates higher probability.

the other two models on the Last.fm dataset. UC slightly outperforms URC for 100 topics. Intuitively, URC captures more of the hidden structure of users' annotation activity in Last.fm. UC also captures the essence of tagging behavior through statistical categorization of tags in latent topics. Contrary, classification of artists based on users' annotation seems to be of inferior quality, probably due to noisy human-provided metadata, which are in their nature, unrestricted, uncontrolled and highly susceptible to personal taste. We conjecture that annotation metadata can be extremely useful in capturing collective knowledge about a domain, such as music genres and artists categorization in Last.fm.

4.4 Recommendation of Social Ties

In this section, we test the effectiveness of our four classification schemes: *a*) Scheme A (Latent Topics & Common Neighbors); *b*) Scheme B (Latent Topics & Shortest Distance); *c*) Scheme C (Latent Topics); and *d*) Scheme D (Ensemble Classification). We randomly split our dataset into two disjoint sets, such that we retain 10%, 25%, 50%, and 75% of the data for training, and the rest for testing. We train UR, UC, and URC models for $T_{UR} = 20$, $T_{UC} = 20$, and $T_{URC} = 50$ hidden topics respectively. The evaluation consists of selecting pairs of users, computing their similarity, and adding links between users in decreasing order of their topical similarity. The pairs of users with highest similarity are those we predict to be most likely tied. We randomly sample $12,716$ pairs of users, out of which 50% are true links and 50% are negative samples. For each predicted social link, we check the actual social network to see if the prediction is correct. We consider Precision, Recall, and the trade–off between them (F–measure) as our performance evaluation metrics.

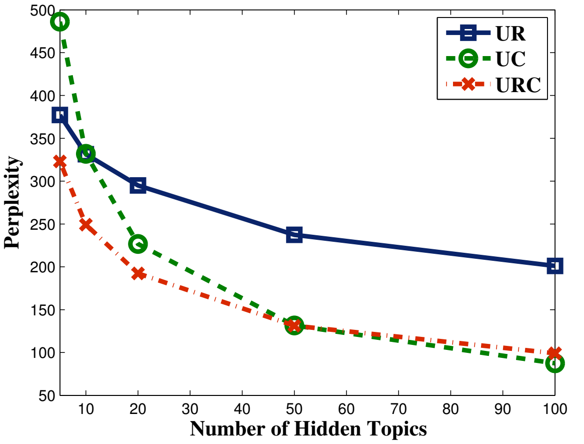

Figure 3: Perplexity for different numbers of hidden topics, for the UR, UC, and URC models.

Figure 4 shows the performance achieved by our classification schemes under our three models with respect to Precision and Recall. We found Scheme B to be the least effective, hence we refrain from discussing its performance any further, even though Scheme B is included in Scheme D, influencing its performance. Scheme B aggregates users' latent similarity with respect to shortest distance, which in effect results in aggregating all training similarity values for true links (i.e. existing social ties) in a single training point in the distance–similarity space. To this extend, the aggregation methodology is non-linear to the preprocessing of true positives and true negatives samples, resulting in considerable loss of information in exchange of scalability gain.

The ensemble achieves the best precision (up to **89.8%** under the UR model), due to its ability to alleviate bad choices made by some of the "expert" classifiers. Even though Scheme D's recall is lower when compared to the rest of the schemes, it is comparable (up to **86.83%** under the UC model) when the training dataset size is small (10%), which would be the case in a real life social network with millions of users. Overall, precision increases or stays constant for dataset size up to 50%, after which point over-fitting occurs. On the other hand, recall drops as a function of dataset size, indicating that small but discriminatory training samples can lead to good performance overall. Ultimately, the trade-off between precision and recall (F-measure) has to be considered for the optimal choice of model, scheme and training dataset size. Of course, different datasets may yield best results for different combinations. The nature and focus of the social network under consideration as well as user-generated content type in this context has to be considered when making this selection.

Support Vector Machines tend to classify every sample to the dominant class under high class imbalance situations, such as in social media, caused by sparsity. To address this problem, we test the performance achieved by our classifiers when calculated separately for the positive and negative classes. Figure 5 shows the results. Intuitively, true negatives are easier to classify correctly under most models, in most cases. Overall, we see a degradation in performance with respect to true positives (which are the hardest to predict) due to over-fitting and noisy observations as

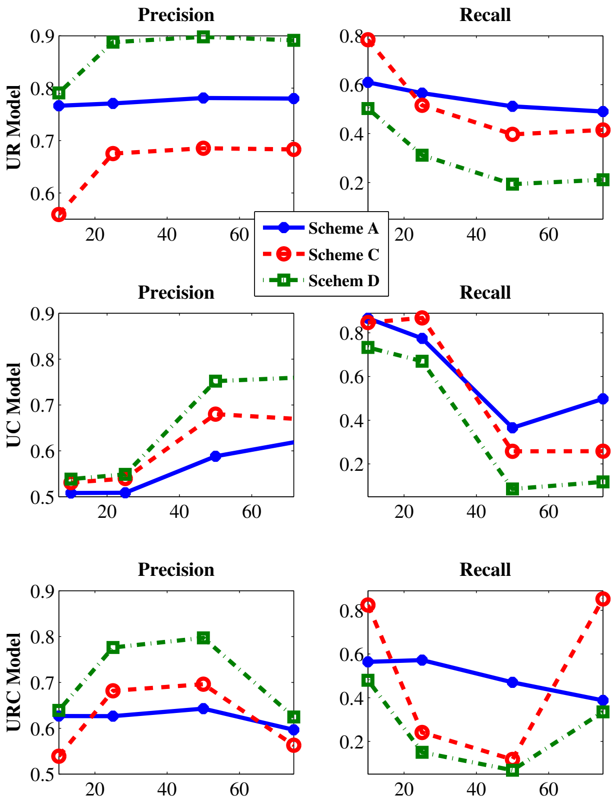

Figure 4: Precision and Recall as a function of training data size. X-axis: Training set size as percentage of complete dataset; Y-axis: Precision/Recall.

the training dataset size increases. Nevertheless, all of our schemes yield reasonable results for practical purposes, for reasonably small training dataset sizes (≤ 20% of complete dataset in all cases).

4.4.1 Comparisons

We compare our schemes with two tag-based similarity metrics, *a*) Cosine Similarity (CS) and *b*) Maximal Information Path (MIP) [29], which have shown superior performance in the content-based network reconstruction task. Scheme D produces class labels, without assigning score values to them, hence we exclude it from our comparison. We have also argued about Scheme B's performance. This leaves us with two Schemes, A and C. We present results in the form of the area under the receiver-operating characteristic curve (AUC). For the calculation of AUC values for the two baselines, we use the complete dataset instead of splitting it into disjoint training and testing sets. This is a conservative strategy which biases the evaluation in favor of the baselines, which have a complete view of the dataset for their similarity calculations.

Figure 6 reports the performance lift of Schemes A and C on the link recommendation task for varying training set size. Lift is defined as % change over best performing baseline, MIP. Positive % change signifies improvement, whereas negative % change indicates superiority of the baseline. The baselines CS and MIP attain AUC values of 0.6087 and 0.6256 respectively. Not all schemes can beat the baseline: our classifiers under the UR and UC model fail to beat the performance of MIP (which is however trained to the complete dataset) when 10% of the data are available for training. In this case the AUC lost from the considerable limitation of training data is minimal, i.e. in the order of 10% or less. The most lift, i.e., % improvement over baseline, is consistently attained by Scheme A, which integrates latent topics with local structural information, under the URC model in all four cases. When 25% or more of the complete dataset are used for training however, both classifiers under any model outperform the baselines.

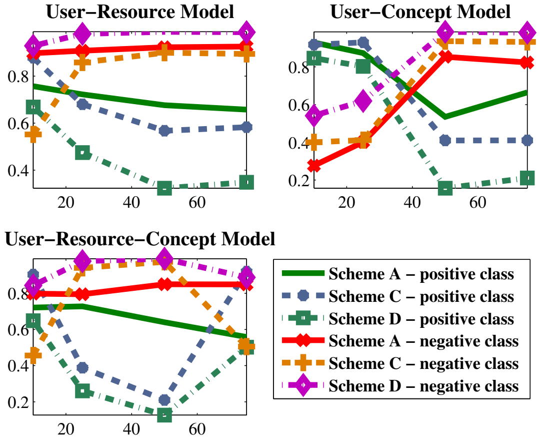

Figure 5: F-measure (calculated for positive & negative classes separately) achieved by classification schemes as a function of training data size. X-axis: Training set size as percentage of complete dataset; Y-axis: F-measure.

5. RELATED WORK

Online social tagging systems have been well studied, leading to a vast literature around this area [7]. Halpin et al. [8] proposed a collaborative tagging model based on preferential attachment and informational value, studying the basic dynamics behind tagging in the social bookmarking site del.icio.us[4]. Others [3, 12] studied personalized tag recommendation and proposed solutions based on relationships between tags and documents. We instead take a probabilistic, generative approach that accurately models the collaborative nature of annotation process overall, generalizing to resources of any type, and annotations.

Recommendation systems based on LDA–like models have been proposed [16, 10, 9]. Instead of treating each recommendation type separately, our models can be effectively used to jointly recommend users, resources, tags and latent topics. Lu et al. [21] proposed a model that represents users, documents, words, and tags, and latent topics and user perspectives in a unified model, but does not capture users' interests. Other models [17, 18] ignore the social aspect of tagging. Instead, by allowing a mixture of users to collab-

[4] `https://delicious.com`

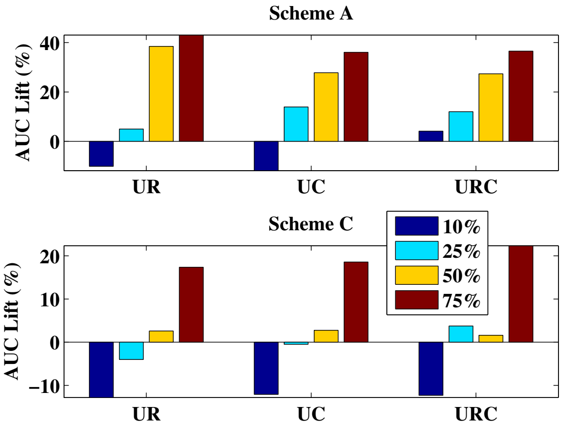

Figure 6: Area under the ROC curve lift achieved by schemes A and C with respect to UR, UC and URC models on the link recommendation task in the Last.fm data set. Lift is defined as % change over MIP baseline.

oratively contribute to the annotation process, we offer a more general solution to a more difficult problem.

The problem of link recommendation for social networks, has been thoroughly investigated. We summarize here the work that is most related to ours. Latent feature based models [11, 23] consider link recommendation as a matrix completion problem and employ latent matrix factorization to learn latent factors for each object, and make predictions. However, such models disregard the local network structure. The URC model (similarly for UR and UC), as an adaptation of the author-topic model, is closely related to methods based on matrix factorization [27]. For applications where models with n-ary relations with $n > 3$ need to be considered, tensor factorization techniques are required [15]. Unfortunately, the straightforward application of higher-order tensor models becomes problematic, due to computational requirements and data sparsity.

Taskar et al. [30] proposed a relational Markov network framework to define a joint probabilistic model over the entire link graph-entity attributes and links, assuming a Markov dependency assumption (the label of one node depends on its neighbors' labels). In contrast to our work, their discriminative model only explains social ties conditioned on the observed variables. Jamali et al. [13] introduced a generalized stochastic block model to predict item ratings and link creation, assuming that a rating network is provided. Topic-link model [19] performed topic modeling and author community discovery in a unified framework, but did not provide reasonable results in the task of link prediction. Pennacchiotti et al. [25] applied LDA for social link recommendation, modeling social media users' streams as documents, represented by words that they emit in the social media. Dietz [5] proposed a generative model that explains artists by tastes that listeners share with their friends. In contrast to our work, their focus was to learn shared topics of interest among friends. Perhaps the work closest to ours is that of Parimi et al. [24]. Their hierarchical system exploits latent user interests based on user profiles, treating users as documents. In this sense, our work is a generalization of their approach, while at the same time requiring significantly less amount of training data to achieve high precision and recall, effectively addressing the high imbalance problem. We further address scalability issues, considering thousands of users who may be arbitrarily connected, resulting in million potential friendship relationships.

6. CONCLUSIONS

In this paper, we investigated three generative probabilistic models of online social tagging systems as a principled way of reducing the dimensionality of this data, capturing at the same time the dynamics of collaborative annotation process. The three models we explored represent users' latent interests over resources and rich metadata describing them. Even though these probabilistic models ignore several aspects of real-world annotation process, such as topic correlation and user interaction, they nonetheless provide a principled and efficient way of understanding user-resource-tag dynamics in very large, online social tagging systems. We showed that in the task of social tie recommendation, a recommendation system that solely considers the topic distribution learned for each user can be outperformed by systems that integrate latent topics and local network structural features. Particularly, we examined four classification schemes in the online music social media site Last.fm, showing how to achieve high recommendation performance. To improve our classifiers' scalability we proposed an aggregation strategy that significantly reduces the number of training samples. In addition to tags, news stories and music artists, there exist other types of resources, metadata and user activities that can be used to further improve recommendation quality. In our future work, we plan to address the challenge of combining multiple heterogeneous information sources within a unified approach. We also plan to establish a mechanism which will automatically identify the most discriminative latent topics and will discard uninformative resources and metadata.

Our results have important implications for the design of social media sites. Our classification schemes can be directly applied to assist users in discovering friends with similar tastes, and form interests groups that are semantically enabled by considering users' latent topical interests, rather than relying on syntactical or frequency-based approaches. Other potential applications not discussed here include, but are not limited to, recommendation of resources and tags to users based on latent semantics, trending topic analysis and trending analysis of users' latent interests, and categorization, classification, and filtering of online information.

7. ACKNOWLEDGMENTS

This work is supported by Chevron Corp. under the joint project, Center for Interactive Smart Oilfield Technologies (CiSoft), at the University of Southern California.

8. REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.

[2] I. Cantador, P. Brusilovsky, and T. Kuflik. 2nd workshop on information heterogeneity and fusion in recommender systems (hetrec 2011). In *Proceedings of*

the 5th ACM conference on Recommender systems, RecSys 2011, New York, NY, USA, 2011. ACM.

[3] E. H. Chi and R. Nairn. Information seeking with social signals: Anatomy of a social tag-based exploratory search browser. In *Workshop on Social Recommender Systems*, SRS ‘10, 2010.

[4] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2010.

[5] L. Dietz. Modeling shared tastes in online communities. In *NIPS Workshop on Applications for Topic Models: Text and Beyond*, 2009.

[6] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235, 2004.

[7] M. Gupta, R. Li, Z. Yin, and J. Han. Survey on social tagging techniques. *SIGKDD Explor. Newsl.*, 12(1):58–72, Nov. 2010.

[8] H. Halpin, V. Robu, and H. Shepherd. The complex dynamics of collaborative tagging. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 211–220, New York, NY, USA, 2007. ACM.

[9] N. Hariri, B. Mobasher, and R. Burke. Context-aware music recommendation based on latenttopic sequential patterns. In *Proceedings of the sixth ACM conference on Recommender systems*, RecSys '12, pages 131–138, New York, NY, USA, 2012. ACM.

[10] M. Harvey, I. Ruthven, and M. J. Carman. Improving social bookmark search using personalised latent variable language models. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 485–494, New York, NY, USA, 2011. ACM.

[11] P. D. Hoff. Multiplicative latent factor models for description and prediction of social networks. *Comput. Math. Organ. Theory*, 15(4):261–272, Dec. 2009.

[12] J. Hu, B. Wang, and Z. Tao. Personalized tag recommendation using social contacts. In *2nd International Workshop on Social Recommender Systems*, SRS ‘11, 2011.

[13] M. Jamali, T. Huang, and M. Ester. A generalized stochastic block model for recommendation in social rating networks. In *Proceedings of the fifth ACM conference on Recommender systems*, RecSys '11, pages 53–60. ACM, 2011.

[14] D. B. Johnson. Efficient algorithms for shortest paths in sparse networks. *J. ACM*, 24(1):1–13, Jan. 1977.

[15] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Rev.*, 51(3):455–500, Aug. 2009.

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

[17] N. Lin, D. Li, Y. Ding, B. He, Z. Qin, J. Tang, J. Li, and T. Dong. The dynamic features of delicious, flickr, and youtube. *J. Am. Soc. Inf. Sci. Technol.*, 63(1):139–162, Jan. 2012.

[18] L. Liu, F. Zhu, L. Zhang, and S. Yang. A probabilistic graphical model for topic and preference discovery on social media. *Neurocomputing*, 95:78–88, Oct. 2012.

[19] Y. Liu, A. Niculescu-Mizil, and W. Gryc. Topic-link lda: joint models of topic and author community. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 665–672, New York, NY, USA, 2009. ACM.

[20] Z. Liu, Y. Zhang, E. Y. Chang, and M. Sun. Plda+: Parallel latent dirichlet allocation with data placement and pipeline processing. *ACM Trans. Intell. Syst. Technol.*, 2(3):26:1–26:18, May 2011.

[21] C. Lu, X. Hu, X. Chen, J.-R. Park, T. He, and Z. Li. The topic-perspective model for social tagging systems. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, pages 683–692, New York, NY, USA, 2010. ACM.

[22] L. Lu and T. Zhou. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications*, 390(6):1150–1170, 2011.

[23] A. K. Menon and C. Elkan. Link prediction via matrix factorization. In *Proceedings of the 2011 European conference on Machine learning and knowledge discovery in databases - Volume Part II*, ECML PKDD'11, pages 437–452, Berlin, Heidelberg, 2011. Springer-Verlag.

[24] R. Parimi and D. Caragea. Predicting friendship links in social networks using a topic modeling approach. In *Proceedings of the 15th Pacific-Asia conference on Advances in knowledge discovery and data mining - Volume Part II*, PAKDD'11, pages 75–86, Berlin, Heidelberg, 2011. Springer-Verlag.

[25] M. Pennacchiotti and S. Gurumurthy. Investigating topic models for social media user recommendation. In *Proceedings of the 20th international conference companion on World wide web*, WWW '11, pages 101–102, New York, NY, USA, 2011. ACM.

[26] J. C. Platt. Advances in kernel methods. chapter Fast training of support vector machines using sequential minimal optimization, pages 185–208. MIT Press, Cambridge, MA, USA, 1999.

[27] M. Rosen-Zvi, C. Chemudugunta, T. Griffiths, P. Smyth, and M. Steyvers. Learning author-topic models from text corpora. *ACM Trans. Inf. Syst.*, 28(1):4:1–4:38, Jan. 2010.

[28] A. Said, E. W. De Luca, and S. Albayrak. How social relationships affect user similarities. In *Workshop on Social Recommender Systems*, SRS ‘10, 2010.

[29] R. Schifanella, A. Barrat, C. Cattuto, B. Markines, and F. Menczer. Folks in folksonomies: social link prediction from shared metadata. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 271–280, New York, NY, USA, 2010. ACM.

[30] B. Taskar, M. fai Wong, P. Abbeel, and D. Koller. Link prediction in relational data. In *in Neural Information Processing Systems*, 2003.