Social Link Prediction in Online Social Tagging Systems

Charalampos Chelmis and Viktor K. Prasanna, University of Southern California

Social networks have become a popular medium for people to communicate and distribute ideas, content, news and advertisements. Social content annotation has naturally emerged as a method of categorization and filtering of online information. The unrestricted vocabulary users choose from to annotate content has often lead to an explosion of the size of space in which search is performed. In this article, we propose latent topic models as a principled way of reducing the dimensionality of such data and capturing the dynamics of collaborative annotation process. We propose three generative processes to model latent user tastes with respect to resources they annotate with metadata. We show that latent user interests combined with social clues from the immediate neighborhood of users can significantly improve social link prediction in the online music social media site Last.fm. Most link prediction methods suffer from the high class imbalance problem, resulting in low precision and/or recall. In contrast, our proposed classification schemes for social link recommendation achieve high precision and recall with respect to not only the dominant class (non-existence of a link), but also with respect to sparse positive instances, which are the most vital in social tie prediction.

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*; H.3.5 [**Information Storage and Retrieval**]: Online Information Services—*Web-based services*; I.2.6 [**Artificial Intelligence**]: Learning

General Terms: Algorithms, Experimentation, Human Factors, Measurement, Performance

Additional Key Words and Phrases: Annotation, collaborative tagging, graphical models, Last.fm, link prediction, link recommendation, machine learning, social bookmarking, social media, topic models, unsupervised learning

ACM Reference Format:
Charalampos Chelmis and Viktor K. Prasanna. 2013. Social Link Prediction in Online Social Tagging Systems. *ACM Trans. Inf. Syst.* V, N, Article A (YYYY), 27 pages.
DOI:http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

Social networking sites have offered Internet users a novel way to organize their online digital content and share content with other users. In general, users of social media sites contribute content which is not restricted to one media type (e.g., documents, photos, URLs). Depending on the social media site, users can annotate content using descriptive text (e.g., title and description of photos in Flickr[1]) or with metadata (i.e., tags). User-generated content mostly comprises of free, unstructured text, which often does not adhere to grammatical and syntactical rules, contains slag terms and abbreviations and is often of restricted length (e.g., 140 characters in Twitter[2]). To improve

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

[2] https://twitter.com/

Author's addresses: C. Chelmis, Department of Computer Science, University of Southern California; V. K. Prasanna, Ming Hsieh Department of Electrical Engineering, University of Southern California.

DOI:http://dx.doi.org/10.1145/0000000.0000000

online content organization, categorization, search and filtering, users have adopted tags (or hashtags). The ability of users to select tags from an unrestricted vocabulary has led to the creation of personalized taxonomies, offering greater malleability and adaptability in information organization than formal classification systems, which impose users with the restriction to annotate content based on predefined keywords. Even though tags tend to be very inconsistent between various users, resulting in a large number of polysemous and synonymous annotations [Golder and Huberman 2006], hierarchies of mediated community knowledge emerge [Lerman and Plangprasopchok 2009].

In current social tagging systems organization, classification and search tend to be rather simplistic in nature, often relying on keyword-based retrieval algorithms or aggregated results stemming from collaborative filtering techniques [Harvey et al. 2011]. Probabilistic models have been successfully used in discovering the set of hidden topics that were responsible for generating a collection of documents (e.g., [Blei et al. 2003]). In this work, we describe three unsupervised models of online social tagging systems as a principled mechanism to address issues of synonymy, polysemy and tag sparseness. Our probabilistic models capture the generative primitives behind online content annotation, while at the same time extract information about users' latent interests and hidden topics from online, large-scale social tagging systems.

Online social media sites users' rich activities reveal crucial information about their behavior and interests. Users' interaction with online content can be effectively captured with tripartite graphs [Halpin et al. 2007]. The models we present in this article mine users' latent interests from their interactions with online content, instead of relying to user-generated profiles, which may be incomplete or obsolete. We benefit from such modeling of users' latent interests into providing answers to a broad range of important queries, such as which users have similar interests (i.e., community detection) and which other users a user would be mostly interested in (i.e., social link prediction). Link prediction in social networks is a challenging problem, as social networking data are inherently noisy and heterogeneous. One key assumption in sociology is the theory of homophily [McPherson et al. 2001], which postulates that people who have similar characteristics tend to form ties. Moreover, it is likely that the stronger the tie, the higher the similarity [Granovetter 1983]. Link prediction models that estimate tie strength from entity attributes and graph structure [Lu and Zhou 2011] or interaction activity and user's profile similarity have been proposed [Xiang et al. 2010]. Such approaches assume the existence of a latent model that captures the causality of the underlying social process by considering relationship strength to be the hidden effect of user profiles similarities and interactions between users. In our work, we first examine which users' activity (annotations, resources, or annotation of resources) is the most discriminative in predicting social ties. We show that users' latent interests can be particularly beneficial to social link prediction and we model the process of social link creation as the hidden effect of such latent profiles combined with network features. Particularly, we propose a framework to integrate our 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.

The main contributions of this work can be summarized as follows. First, we propose a novel generative modeling of tripartite graphs in social media sites with three probabilistic models that simultaneously capture users' interests with respect to annotation of resources and hidden topics. We provide a systematic comparison of our models in the task of uncovering hidden topics and we illustrate numerous applications. Consequently, we propose several scalable methods for learning to classify social links, based on latent semantics and local network structure. We compare our methods against state of the art social link prediction techniques on a real-world dataset.

The outline of the article is as follows: Section 2 describes the basic structure of tripartite graphs and Section 3 introduces our three probabilistic models for tripartite graph generation. Section 4 briefly describes a social link prediction technique based on semantic similarity of user-generated metadata. We use this technique as a baseline to demonstrate the effectiveness of our proposed models in the task of social tie recommendation. Section 5 discusses our four scalable social link classification schemes, which exploit latent semantics and local network structure. Section 8 evaluates our clustering schemes on a real-world dataset and contrasts it to other approaches. Section 9 summarizes previous work, while Section 10 concludes with a discussion of the implications of our findings and directions of future work.

2. STRUCTURE OF TRIPARTITE GRAPHS

A social network is often represented as *sociogram* [Wasserman and Faust 1994], in which nodes represent users and arcs represent explicit relationships between them. A sociogram is realized as *graph*, *adjacency matrix* or distributed *adjacency lists* (each node in the network maintains a local collection of its neighboring vertices). In order to exploit implicit relationships between users, *tripartite graph* models have also been proposed [Halpin et al. 2007], as shown in Figure 1.

Tripartite graphs offer a mechanism to describe and capture users' behavior and interests in terms of their activities. A tripartite graph is a graph whose vertices can be divided into three disjoint 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., photos) $\mathcal{R} = \{r_1, ..., r_R\}$. A resource $r \in \mathcal{R}$ is annotated with a set of concepts $\mathbf{c}_r \in \mathcal{C}$ of size N_r (similarly created, used, bookmarked or shared), by a set of actors $\mathbf{a}_r \in \mathcal{A}$. 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. It is possible to cluster vertices that belong to any of the three disjoint sets of a tripartite model so as to extract emergent semantics. Tripartite graphs can this way be reduced into three bipartite graphs, which model associations between actors and concepts (bipartite graph AC), concepts and resources (bipartite graph CR), and actors and resources (bipartite graph AR). Bipartite graphs are easier to comprehend and work with but the reduction process discards higher dimensional links between the three sets, which could otherwise be extremely useful in the analysis of the social network at hand. A bipartite graph can be further reduced to produce two simple, weighted graphs. For example, the bipartite graph of actors and concepts (AC) may be reduced into two graphs, one for actors (graph A) and one for concepts (graph C). In this case, the reduced graph A models relationships between actors, weighted by the number of times two actors have used same concepts.

The creator of a resource is often considered to be its owner, but many actors may use, bookmark or share a resource, thus becoming “owners” themselves. Further, many actors may collectively annotate a resource, socially contributing to its set of concepts. We consider artists in Last.fm[3] as resources, which are annotated with tags. Tags become concepts in our modeling. More complex hierarchical Bayesian models can be designed if more types of resources and concepts are considered. The models we describe below can be naturally extended to accommodate other resources and annotation types, such as annotations of Flickr photos, or descriptive text of Youtube videos.

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

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

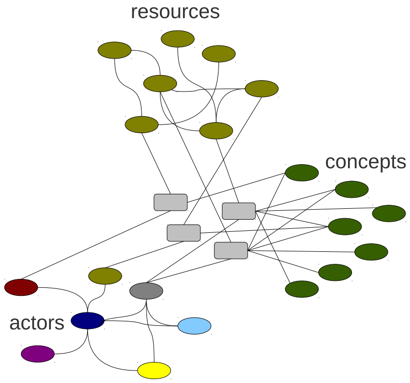


Fig. 1: Tripartite graph model of a social network.

with different keywords, which may have similar meaning (e.g., synonyms) or with linguistic variations of the same keyword due to the uncontrolled vocabulary (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 i-phone product. We use probabilistic models as a mechanism to address such issues of synonymy, polysemy and tag sparseness and effectively model tripartite graphs in order to capture users’ interests in social networks.

3. GENERATIVE MODELS OF TRIPARTITE GRAPHS IN SOCIAL NETWORKS

3.1. The User-Resource-Concept Model

We introduce User-Resource-Concept model (**URC**), a probabilistic author-topic model [Rosen-Zvi et al. 2010] to model users’ interests based on their resource usage and annotation behavior. 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. The process of resource annotation can be described as a stochastic process. A group of users $\mathbf{a}_r$, which for the purposes of estimation we assume is observed, collectively annotate resource r. For each resource annotation a user a is chosen uniformly at random. Based on user a’s interests and the nature of the resource, a set of topics is selected. Concept c_{ri} (e.g., tag) is generated based on the selected set of topics.

This generative process is described in graphical form in Figure 2a. x indicates the user, chosen from $\mathbf{a}_r$, responsible for a given annotation. 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 actor a. We use θ_a to denote the a^{th} column of the matrix. The mixture weights for 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 distribution over words associated with each topic. ϕ_t represents the probability of generating concepts from topic t. Table I summarizes this notation. To summarize, we have the following data generation process for URC:

For each actor $a \in \mathcal{A}$ choose $\theta_a \mid \alpha \sim$ Dirichlet(α).
For each topic $t \in T$ choose $\phi_t \mid \beta \sim$ Dirichlet(β).

Table I: Notation

Set of actors	$\mathcal{A}$	Set
Number of unique actors	A	Scalar
Set of concepts	$\mathcal{C}$	Set
Number of unique concepts	C	Scalar
Total number of concepts	N	Scalar
Set of resources	$\mathcal{R}$	Set
Number of unique resources	R	Scalar
Number of topics	T	Scalar
Dirichlet prior	α	Scalar
Dirichlet prior	β	Scalar
Probabilities of concepts given topics	Φ	$C \times T$ matrix
Probabilities of concepts given topic t	ϕ_t	C-dimensional vector
Probabilities of topics given actors	Θ	$T \times A$ matrix
Probabilities of topics given actor α	θ_α	T-dimensional vector
Number of actors associated with the r^{th} resource	A_r	Scalar
Actors related to the r^{th} resource	$\mathbf{a}_r$	A_r-dimensional vector
Number of concepts associated with the r^{th} resource	N_r	Scalar
Concepts related to the r^{th} resource	$\mathbf{c}_r$	N_r-dimensional vector
i^{th} concept in the r^{th} resource	c_{ri}	i^{th} component of $\mathbf{c}_r$
Concepts related to all resources	$\mathbf{c}$	N-dimensional vector
Actor assignments	$\mathbf{x}$	N-dimensional vector
Actor assignments for concept c_{ri}	x_{ri}	i^{th} component of $\mathbf{x}_r$
Topic assignments	$\mathbf{z}$	N-dimensional vector
Topic assignments for concept c_{ri}	z_{ri}	i^{th} component of $\mathbf{z}_r$

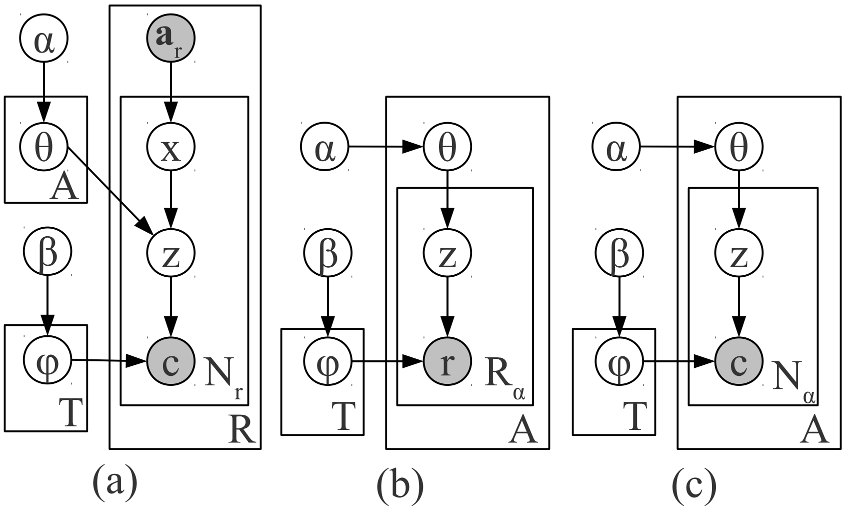


Fig. 2: Generative models of tripartite graphs. (a) User-Resource-Concept model, (b) User-Resource model, (c) User-Concept model.

```
For each resource $r \in R$, given actors vector $\mathbf{a}_r$,
    For each concept $i \in N_r$
    Choose actor $x_{ri} \mid \mathbf{a}_r \sim \text{Uniform}(\mathbf{a}_r)$
    Choose topic $z_{ri} \mid \theta_{x_{ri}}, x_{ri} \sim \text{Multi}(\theta_{x_{ri}})$
    Choose concept $c_{ri} \mid z_{ri}, \beta \sim \text{Multi}(\phi_{z_{ri}})$.
```

The joint distribution of observed and hidden variables is:

$$\begin{aligned} P(\mathbf{c}, \mathbf{z}, \mathbf{x}, \Phi, \Theta \mid \alpha, \beta, \mathcal{A}) \;=\; & \prod_{t=1}^{T} P(\phi_t \mid \beta) \prod_{a=1}^{A} P(\theta_a \mid \alpha) \\ & \prod_{r=1}^{R} \prod_{i=1}^{N_r} P(x_{ri} \mid \mathbf{a}_r) P(z_{ri} \mid \theta_a, x_{ri}) P(c_{ri} \mid \phi_t, z_{ri}). \end{aligned} \tag{1}$$

3.2. The User-Resource Model

The User-Resource Model (**UR**), shown in Figure 2b, is a simplification of the URC model and is structurally equivalent to the LDA model [Blei et al. 2003]. We begin by reducing the tripartite graph of users, resources and concepts into a bipartite graph of users and resources. In this modeling, users' interests are expressed in terms of activity involving similar resources (e.g., users A and B have similar tastes if user A creates a resource R, which user B comments on). Hence, each user owns one "document", and resources become vocabulary terms that users select to "compose" their documents.

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. To summarize, the UR model assumes the following generative process for each actor $a \in \mathcal{A}$:

```
Choose $\theta \mid \alpha \sim$ Dirichlet($\alpha$).
For each topic $t \in T$ choose $\phi_t \mid \beta \sim$ Dirichlet($\beta$).
For each resource $r_i \in R_a$,
    Choose topic $z_i \mid a \sim$ Discrete($\theta$)
    Choose resource $r_i \mid z_i, \beta \sim$ Discrete($\phi_{z_i}$).
```

The joint distribution of observed and hidden variables in this case is:

$$P(\mathbf{r}, \mathbf{z}, \Phi, \Theta \mid \alpha, \beta) = P(\theta \mid \alpha) \prod_{t=1}^{T} P(\phi_t \mid \beta) \prod_{i=1}^{R_a} P(z_i \mid \theta) P(r_i \mid \phi_t, z_i). \tag{2}$$

3.3. The User-Concept Model

The User-Concept (**UC**) model is shown in Figure 2c. Similarly to UR model, this too is a simplification of the URC model. UC is an adaptation of the LDA model [Blei et al. 2003] with the difference that users are modeled based on their tag usage. In order to construct this model, we aggregate annotations assigned by users to resources they "own" and use these tags as vocabulary terms. The motivation for this reduction stems from our analysis of tripartite graphs' structure in Section 2. There we argued that bipartite graphs are easier to work with, even though they discard information that could otherwise be used to enhance the modeling of users' online activities. We use this model as a simpler and more scalable solution to our problem, and compare its effectiveness against URC.

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. To summarize, the UC model assumes the following generative process for each actor $a \in \mathcal{A}$:

Choose $\theta \mid \alpha \sim$ Dirichlet(α).
For each topic $t \in T$ choose $\phi_t \mid \beta \sim$ Dirichlet(β).
For each concept $c_i \in N_a$,
    Choose topic $z_i \mid a \sim$ Discrete(θ)
    Choose concept $c_i \mid z_i, \beta \sim$ Discrete(ϕ_{z_i}).

The joint distribution of observed and hidden variables in this case is:

$$P(\mathbf{c}, \mathbf{z}, \Phi, \Theta \mid \alpha, \beta) = P(\theta \mid \alpha) \prod_{t=1}^{T} P(\phi_t \mid \beta) \prod_{i=1}^{N_a} P(z_i \mid \theta) P(c_i \mid \phi_t, z_i). \tag{3}$$

3.4. Parameter Estimation

Given any one of the three models we described above, we can obtain information about which topics users are mostly interested in, as well as a representation of annotations with respect to these topics, by estimating parameters Φ (probability of topics given concepts) and Θ (probability distribution over topics for each user, given concepts). The hidden structure of topics is captured by the posterior distribution of the hidden variable $\mathbf{z}$ (probability of topic mixtures of concepts). We adopt collapsed Gibbs sampling [Griffiths and Steyvers 2004] to compute the posterior distribution on $\mathbf{z}$ and then use the result to infer matrices Φ and Θ.

4. SOCIAL LINK PREDICTION USING HIDDEN TOPICS (SLIGHT)

One natural application of our modeling is social link prediction given some snapshot of a tripartite graph. Makrehchi [2011] constructed a semi-bipartite graph of extracted hidden topics from user profiles and then applied topological metrics such as Katz [Katz 1953] and short path scores to rank and recommend users. Makrehchi [2011] showed that this method outperforms approaches that rely on similarity measures of feature vectors (i.e., Bag of Words) and low rank approximation (i.e Latent Semantic Indexing (LSI)). We extend this approach by considering resources and metadata to represent users' interests instead of documents consisting of words. Our goal is to use this approach as a baseline in comparison to our novel techniques for social link prediction (see Section 5) in a generic social network that is not as focused as academic networks extracted from technical paper co-authorships.

Gibbs sampling of the posterior distribution on z results into generating matrices Φ, Θ and $\mathbf{C}$ [Rosen-Zvi et al. 2010]. Topic-actor matrix Θ in particular represents a bipartite graph linking topics to actors. Using matrix Θ, we can build a semi-bipartite graph $\mathbf{G}$ [Makrehchi 2011] of size $(A+T) \times (A+T)$:

$$\mathbf{G} = \begin{bmatrix} \mathbf{S} & \Theta^\top \\ \Theta & \Theta \times \Theta^\top \end{bmatrix}. \tag{4}$$

$\mathbf{S}$ represents relationships between users and is unknown. Makrehchi [2011] used Katz score [Katz 1953] to predict the missing values of the unknown block $\mathbf{S}$, such that $\mathbf{S} = Katz(G)$, in an academic social network. Katz score, a generalization of degree centrality, measures the degree of influence of an actor in a social network [Katz 1953]. Typical centrality measures only consider the geodesic distance between a pair of actors. Instead, Katz takes into account the total number of walks between a pair of actors, penalizing long paths by an attenuation factor $\delta \in (0, 1)$ (typically the spectral norm of matrix $\mathbf{G}$), raised to the power of path length. The Katz score for any two entries g_i, g_j of matrix $\mathbf{G}$ can be computed as follows:

$$Katz(g_i, g_j) = \sum_{l=1}^{\infty} \delta^l |path_l(g_i, g_j)|, \tag{5}$$

where $path_l(g_i, g_j)$ is the set of all paths of length l between g_i, g_j.

4.1. Threshold Selection

Due to the fact that link prediction between two nodes is a binary classification problem, similarity matrix **S** has to be converted into a binary adjacency matrix. The process consists of examining the similarity between each pair of users and checking if its value exceeds a threshold. As the similarity threshold decreases, more links are added, leading to more true positives but also to more false positives. We determine the best threshold value automatically based on the probability of the existence of a link in a social network [Makrehchi 2011]. In a sparse, directed social network, the number of existing links is considerably smaller than all possible links. The density of a directed social network can be calculated as $\Delta = \frac{L}{n(n-1)}$, where L is the number of true links in the network and n is the number of nodes. The higher the density of the graph, the higher the probability of a link, hence the higher the probability that nodes are connected to others. Conversely, low density indicates a sparse graph with few connected nodes, isolated communities and unreachable nodes.

Given link probability $p = \Delta$, we determine the optimal threshold value τ by minimizing the squared error between the empirical density of the graph that results from link prediction by converting all similarity values that exceed τ into links, and the true density of the graph p. Formally, we define the optimal threshold as follows:

$$\hat{\tau} \doteq \min_{\tau} \left\{ \left(\left[\frac{\sum_{i=1}^{n} \sum_{j=1}^{n} \mathbb{1}\{S(n_i, n_j)\}}{n(n-1)} \right] - p \right)^2 \right\}, \tag{6}$$

where

$$\mathbb{1}\{S(n_i, n_j)\} = \begin{cases} 0, & S(n_i, n_j) \leq \tau \\ 1, & S(n_i, n_j) > \tau \end{cases}. \tag{7}$$

5. SOCIAL LINK PREDICTION USING LATENT SEMANTICS AND NETWORK STRUCTURE

Social networking users involve in rich activities that reveal crucial information about their interests and tastes. Explicit user profiles, typically consisting of personal information like hobbies, favorite movies and music, etc., can be mined in order to identify user interests, based on which friendship predictions can be made. However, information in user profiles tends to be scarce or obsolete. Instead of mining explicit user profiles, we gather valuable information about users' interests from metadata that describe their social network activities. In Last.fm we capture music genre preferences by mining listening frequencies to artists as well as by recording tags, with which users annotate artists they are mostly listening to. Schifanella et al. [2010] showed that even though there is no globally shared vocabulary in Flickr, high vocabulary similarity between users suggests the presence of a link between them.

In our work, we exploit the latent description of users' interests (matrix Θ), which we learn using Gibbs sampling. Our intention is to explore whether the activation of a social link induces a local alignment of interests or if conversely a similarity in interests triggers the creation of a social link. We test this hypothesis on our Last.fm dataset, which provides annotation metadata needed to construct our generative models, as well as ground truth social network to evaluate the accuracy of our recommendations.

We describe user interests in a latent space, organizing them in topics, emerging from user activity and annotation process. To do so, we use the generative models we presented in Section 3, treating users as authors and annotations as the vocabulary authors use to describe resources. Since we do not *a* priori know what is the optimal number of topics, we vary the number of topics achieving description of user interests in variable granularities, from more abstract to extremely specific. We treat link recommendation as a binary classification problem, where 1 indicates a link and 0 indicates the absence of a link.

In the rest of this section we propose four classification schemes that utilize matrix Θ to learn how to recommend appropriate links. All classifiers are generated as support vector machines (SVM) with Gaussian radial basis function kernels [Cristianini and Shawe-Taylor 2010]. Finally, the last classification scheme exploits all previous classifiers building a hierarchical system.

5.1. Latent Topics & Common Neighbors Scheme

Many social link prediction approaches calculate graph-based proximity scores [Lu and Zhou 2011], asserting that the "closer" two nodes 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 them. Proximity metrics used in prior work include neighborhood based methods and methods based on the ensemble of all paths [Lu and Zhou 2011].

Neighborhood based methods, such as the number of common neighbors, the Jaccard coefficient, which computes the probability of two users sharing neighbors, and Adamic/Adar, which refines the simple counting of common neighbors by weighting rarer neighbors more heavily, exploit local network features. For simplicity and computational efficiency, we use the number of common neighbors between two users as a prominent indicator of social link creation. 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 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:

$$\sigma(u,v) = \frac{\sum_t \Theta(t,u)\Theta(t,v)}{\sqrt{\sum_t \Theta(t,u)^2}\sqrt{\sum_t \Theta(t,v)^2}}. \tag{8}$$

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:

$$F(u,v) = [\sigma(u,v), CN(u,v)]\,. \tag{9}$$

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 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:

$$avg_\sigma(k) = \frac{1}{|p : k_p = k|} \sum_{p:k_p=k} \sigma(p), \tag{10}$$

where p denotes a user pair (u, v) and k_p denotes the number of common neighbors for user pair p.

5.2. Latent Topics & Shortest Distance Scheme

Instead of using the number of common neighbors, here, we use shortest distance to capture graph based similarity between users u and v, denoted as $SD(u, v)$. The feature vector for a user pair (u, v) is therefore constructed as:

$$F(u, v) = [\sigma(u, v), SD(u, v)]\,. \tag{11}$$

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 (10), with the difference that in this case k_p denotes the shortest distance value for user pair p.

5.3. Latent Topics Classification Scheme

Here we focus solely on similarity of users' interests, 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. Again, we consider column $\Theta(:, u)$ as feature vector for user u and we compute the pointwise squared distance between feature vectors of users u and v. The feature vector for a user pair (u, v) is therefore constructed as:

$$F(u, v) = \left[(\Theta(1, u) - \Theta(1, v))^2, \ldots, (\Theta(T, u) - \Theta(T, v))^2\right]. \tag{12}$$

$F(u, v)$ 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. Note that the optimization objective of this classifier is to minimize the distance between users u and v between whom a tie exists. In contrast, the two previous schemes assume maximum similarity values between such users.

5.4. Ensemble Classification Scheme

The first step in an ensemble approach is data partitioning. Each partitioning technique should have a unique view of the data or use a different underlying model to generate the data partitions. In our approach, we select classifiers that partition the data using different set of features and appropriate similarity metrics discussed in the previous subsections. In particular, we train each of the above three classifiers individually using the same set of training data. This results in classifiers Cl_1, Cl_2, and Cl_3 respectively.

We combine the predictions of each classifier using a consensus mechanism, according to which each classifier is treated as expert casting a vote for or against the existence of a link between a pair of users. We set Cl_1, Cl_2 and Cl_3's ensemble weights to equal values and we normalize them such that $\sum\limits_{i=1}^{3} \lambda_{Cl_i} = 1$. 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:

$$\xi_{Cl_i}(p) = \begin{cases} 1, & \exists\, e(u, v) \\ 0, & otherwise \end{cases}, \tag{13}$$

where $e(u, v)$ denotes a directed edge between users u and v. We compute the consensus score for p as $\sum\limits_{i=1}^{3} \lambda_{Cl_i} \xi_{Cl_i}(p)$. We could have learned different weights for each classifier, indicating our confidence in its predictions. However, this procedure imposes another

Table II: Symbols used in Complexity Analysis

$\mathbf{G}$	Social Network
Λ	Adjacency matrix of $\mathbf{G}$
E	Number of edges in $\mathbf{G}$
A	Number of users
V	Vocabulary size
U_{max}	Maximum number of users that can be associated with a resource
A^{Train}	Training set size

round of supervised training phase, which would unnecessarily increase the complexity of our approach. In our evaluation section we show that, despite its simplicity, the majority voting scheme is quite effective in producing high quality recommendations.

5.5. Complexity Analysis

We performed our experiments on a 2.4 GHz Intel Core 2 Duo, with 2 GB of memory, running Windows 7. For our evaluation, we used a real-world dataset (see Section 6) of 2K users from Last.fm online music system [Cantador et al. 2011]. All algorithms were implemented in Matlab. We now discuss in detail the computational complexity of our approach and examine its ability to scale into large datasets. Table II summarizes the symbols used in our analysis.

5.5.1. Complexity of Inferencing Latent Models. The worst case time complexity of each iteration of the Gibbs sampler is $O(VU_{max}A)$. As complexity is linear in V, Gibbs sampling can be efficiently carried out on large data sets [Rosen-Zvi et al. 2010]. Considerable speedup gains can be achieved by optimizing Gibbs sampling and by successfully incorporating recent advances in parallel and cloud computing [Liu et al. 2011].

5.5.2. Complexity of Structural Features Calculation. Next, we discuss the computational complexities of graph-based similarity metrics.

Common Neighbors. Naively, Λ^2 computes CN for all user pairs. Intuitively, $\Lambda^2(u, v)$ denotes the number of different length 2 paths that connect users (u, v). Multiplication of extremely sparse matrices (i.e., adjacency matrix) is inefficient and can become very expensive for large datasets. Instead of using matrix multiplication in calculating 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 $O(Ak^2)$.

Shortest Distance. We find the shortest path SD between any two users using Johnson's algorithm [Johnson 1977], resulting in a time complexity of $O(AlogA + AE)$. A faster implementation based on a min-priority queue (i.e., Fibonacci heap) can further reduce running time to $O(AlogA + E)$.

5.5.3. Complexity of Averaging Strategy. To reduce the number of training samples provided to our SVM classifiers, we first average similarity values over the number of common neighbors (similarly for shortest distance) as shown in Equation (10). 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 $O(KA|S_{CN_k}|)$, 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.

5.5.4. Complexity of SVM Classification. Support vector machines (SVMs), though accurate, are not preferred in applications requiring great classification speed due to the number of support vectors being large. Standard SVM training requires the solution of a very large quadratic programming (QP) optimization problem, which directly involves inverting the kernel matrix, resulting in $O(A^{Train^3})$ time and $O(A^{Train^2})$ space complexities [Keerthi et al. 2006]. Due to our averaging strategy, A^{Train} is already sufficiently small (i.e., $A^{Train} \ll A$). However, one hardly ever needs to estimate the optimal solution, and the training time for a linear SVM to reach a certain level of generalization error actually decreases as training set size increases [Shalev-Shwartz and Srebro 2008]. Tsang et al. [2005] proposed an approximation algorithm that obtains approximately optimal solution, while at the same time having a time complexity that is linear in A^{Train} and a space complexity that is independent of A^{Train} for nonlinear kernels.

In our work, we use Sequential Minimal Optimization (SMO) to train our SVM classifiers [Platt 1999]. SMO divides the quadratic programming optimization problem into smaller problems that can be solved analytically. Further, as SMO memory requirements grow linearly to the training set size, SMO can handle very large training sets [Platt 1999]. In testing time, we need to pass a user pair instance onto an SVM model to find the hypothesis with the highest confidence (i.e., existence of a link or not). The time complexity for this step is $\sim O(1)$ (linear to the number of the support vectors and linear to the number of features).

6. DATASET

To examine the effectiveness of our models (see Section 3) and classification schemes (see Section 5), we use a dataset containing social networking, tagging and music artist listening information from a set of 2K users from Last.fm online music system [Cantador et al. 2011]. Last.fm builds profiles of each user's musical tastes by recording details of the songs users listen to. Further, Last.fm allows users to create social networks by listing friends (users who have similar musical tastes to them).

Our Last.fm dataset consists of 1,892 users with 25,434 directed user friend relations, 17,632 artists and 92,834 user-listened artist relations, i.e., tuples of the form <user, artist, listening count>. Further, the dataset contains 11,946 unique tags, which were used in 186,479 annotations, i.e., tuples of the form <user, tag, artist>. This leads to a vocabulary size of $R = 17,632$ in our User-Resource model and $C = 11,946$ unique words in our User-Concept and User-Resource-Concept models for this dataset. We 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.

6.1. Predictive Power

To demonstrate the effectiveness of our generative models on uncovering hidden topics, we compute their perplexity [Rosen-Zvi et al. 2010] (i.e., their ability to predict tags or artists for new users). 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 three models' perplexity scores on varying number of hidden topics.

URC yields lower perplexity overall than the other two models on the Last.fm dataset. UC slightly outperforms URC for 100 topics. UR and UC models can be seen as extensions of the classic LDA model, whereas URC is an extension of the Author-Topic model. 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 solely on users' annotation seems to be of inferior quality, probably due to noisy human-provided

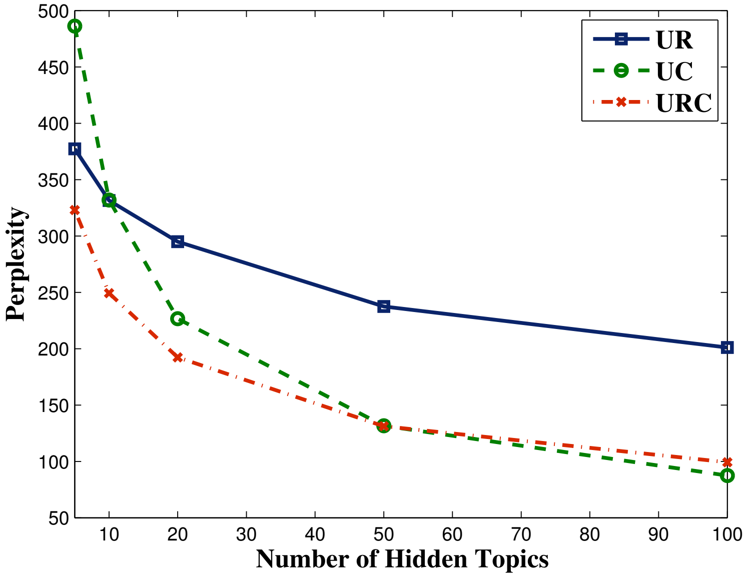


Fig. 3: UR, UC and URC perplexity for varying number of hidden topics.

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 in Last.fm.

6.2. Examples of Topic and User Distributions

In this section, we provide illustrative examples of topics learned by our three models on our Last.fm dataset. The topics are extracted from a single sample at the 2000th iteration of the Gibbs sampler. Figure 4 shows 4 topics (out of 50) learned by the URC model. Each topic is illustrated with (a) the top 10 tags most likely to be generated conditioned on the topic and (b) the top 10 most likely users to have generated a tag conditioned on the topic. Users' identities have been anonymized for privacy, offering no particular insights to our analysis. We include a sample here for completeness, while we refrain from listing users' probabilities for UR and UC models for space efficiency. Figure 5 shows 4 topics (out of 50) learned by the UR model along with a list of the top 10 artists most likely to be generated conditioned on the topic. Figure 6 shows 4 topics (out of 50) learned by the UC model. Each topic in this case is illustrated with the top 10 tags most likely to be generated conditioned on the topic.

Topics learned by the URC model offer a qualitative representation of music genres in Last.fm, "generating" a music taxonomy based on user-specific tags. The top 10 most likely artists in each topic learned by URC are well-known in terms of popularity and fame. Solo artists and music bands are being categorized in corresponding music categories in this case. Finally, even though most of the topics in our models semantically capture music genres, some topics illustrate some other types of discovered themes. For instance, topic 5 in UC captures users' preferences in the form of explicitly stated feelings and opinions with respect to specific artists. Notably, URC topics 44 and 47 match surprisingly well UC topics 47 and 45 accordingly.

Topic 3		Topic 16		Topic 44		Topic 47	
Tag	Prob	Tag	Prob	Tag	Prob	Tag	Prob
latin	0.14264	classic rock	0.31569	80s	0.34940	heavy metal	0.18222
spanish	0.08686	70s	0.14149	new wave	0.20432	progressive metal	0.11349
latin pop	0.04755	60s	0.10063	british	0.05184	metal	0.10689
fantastic	0.03338	rock	0.09980	post-punk	0.02647	power metal	0.09490
ballad	0.03201	80s	0.03354	english	0.01728	melodic death metal	0.06555
english	0.03109	oldies	0.02816	uk	0.01467	symphonic metal	0.05846
cool as	0.02835	rock n roll	0.02650	scottish	0.01193	gothic metal	0.04427
tinosoft	0.02378	rock and roll	0.02595	80s pop	0.01152	speed metal	0.03131
oh so catchy	0.02241	guitar	0.01974	college rock	0.01125	finnish	0.01884
good old times	0.02103	southern rock	0.01601	80's	0.00946	folk metal	0.01884
User	Prob	User	Prob	User	Prob	User	Prob
user_1974	0.12323	user_608	0.10791	user_224	0.08240	user_715	0.09009
user_1283	0.06793	user_699	0.08451	user_1202	0.07601	user_546	0.07456
user_1171	0.03889	user_1702	0.05678	user_1209	0.07270	user_1860	0.06415
user_2071	0.03232	user_541	0.05002	user_423	0.03596	user_2053	0.03737
user_1245	0.03005	user_282	0.04214	user_1210	0.03309	user_177	0.03498
user_396	0.01944	user_1271	0.03327	user_117	0.03243	user_1623	0.03361
user_49	0.01136	user_1846	0.03327	user_1038	0.03221	user_978	0.03037
user_871	0.00783	user_1792	0.03105	user_1958	0.03023	user_377	0.02576
user_21	0.00682	user_1086	0.02451	user_2014	0.02625	user_595	0.02047
user_1662	0.00505	user_1249	0.02063	user_1900	0.02603	user_235	0.01723

Fig. 4: Top tags and users for 4 topics (out of 50) learned by the URC model.

Topic 1		Topic 4		Topic 21		Topic 22	
Artist	Prob	Artist	Prob	Artist	Prob	Artist	Prob
Avril Lavigne	0.37408	Pink Floyd	0.26757	Britney Spears	0.64725	Iron Maiden	0.23771
Lifehouse	0.05404	Led Zeppelin	0.09341	Hilary Duff	0.01994	Megadeth	0.04520
SHINee	0.04435	The Beatles	0.02897	Lindsay Lohan	0.01652	Dream Theater	0.03659
Daughtry	0.03399	The Doors	0.02618	Christina Aguilera	0.01593	Metallica	0.02302
Enrique Iglesias	0.01537	Tangerine Dream	0.02544	Madonna	0.01586	50 Cent	0.02217
Takanashi Yasuharu	0.01319	Electric Light Orchestra	0.01970	Avril Lavigne	0.01415	Sonata Arctica	0.02130
Shayne Ward	0.01293	The Who	0.01942	Ashlee Simpson	0.01148	Black Sabbath	0.01974
Westlife	0.01280	Deep Purple	0.01820	Kylie Minogue	0.01049	Judas Priest	0.01889
Mando Diao	0.01198	Frank Zappa	0.01654	Ashley Tisdale	0.00914	Dio	0.01774
Bowling for Soup	0.01182	Camel	0.01263	Ke$ha	0.00894	Blind Guardian	0.01695

Fig. 5: Top artists for 4 topics (out of 50) learned by the UR model.

Topic 5		Topic 8		Topic 45		Topic 47	
Tag	Prob	Tag	Prob	Tag	Prob	Tag	Prob
beautiful	0.17764	uk	0.11470	heavy metal	0.13581	new wave	0.22783
awesome	0.09068	usa	0.09936	thrash metal	0.09054	80s	0.16628
sad	0.06328	00s	0.08158	progressive metal	0.07005	post-punk	0.12037
dreamy	0.05380	noise	0.05997	metal	0.06797	punk	0.06155
brilliant	0.03688	post rock	0.03765	hard rock	0.06770	synth pop	0.04815
atmospheric	0.03384	rock	0.03312	death metal	0.06327	new romantic	0.03524
favourites	0.03147	alternative	0.02580	power metal	0.05372	college rock	0.02035
melancholy	0.02437	experimental	0.02545	melodic death metal	0.05275	rockabilly	0.01762
best christmas songs	0.01929	punk	0.02162	black metal	0.03890	goth	0.01737
genius	0.01895	england	0.02092	symphonic metal	0.03115	album rock	0.01216

Fig. 6: Top tags for 4 topics (out of 50) learned by the UC model.

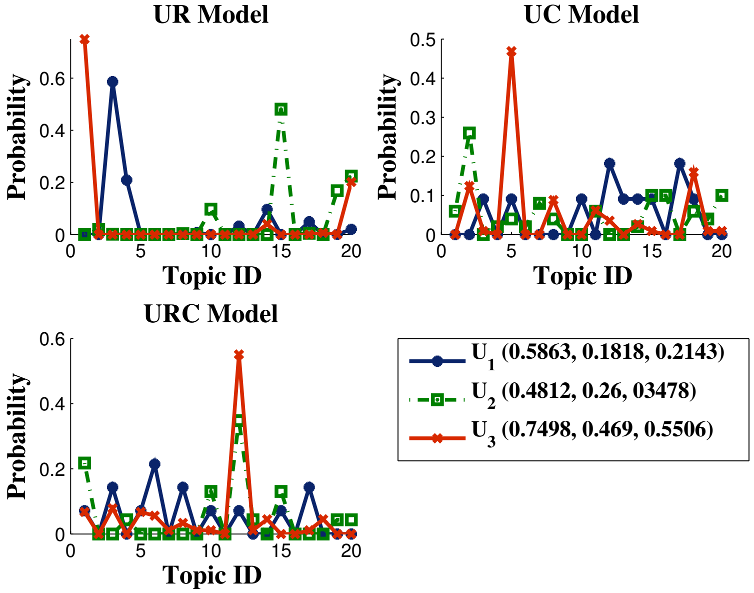


Fig. 7: Probability distribution of three most popular users' latent interests over twenty topics.

6.3. User Focus Analysis

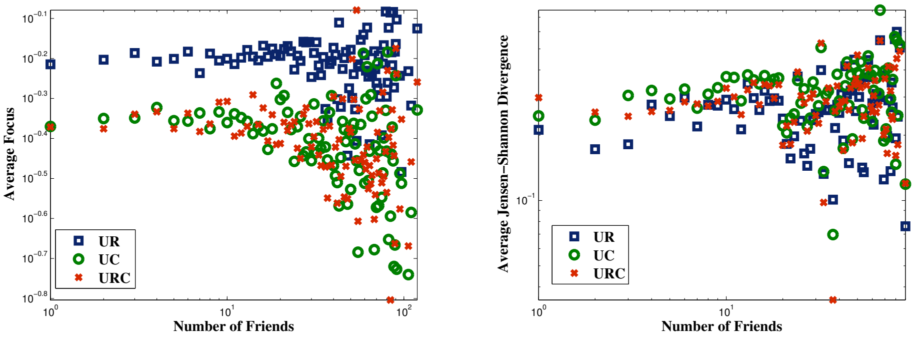
Our generative models capture latent users' interests in different contexts. A latent interest profile can be built for each user for each of our models, facilitating quantitative measurement of user "focus". We measure the "focus" of a user u to characterize dispersion of user latent interests across multiple topics. To measure u's focus, we first sort u's topic probability vector in descending order and then sum the difference between topic pairs. Formally, we define user "focus" as:

$$f(u) \doteq \sum_{t=1}^{T-1} (u_{p_t} - u_{p_{t+1}}), \tag{14}$$

where u_{p_t} denotes the probability of topic t for user u. Intuitively, a perfectly "focused" user has a focus value of one, whereas the focus of a completely "diverse" user is equal to zero.

Figure 7 shows the probability distribution of three most popular users' latent interests over twenty topics, for each of our models. Users' focus values for each model are provided inside parenthesis in the legend. User U_3 exhibits more focused interests than the rest two users in all three models, whereas U_1 demonstrates clear focus only under the UR model. This indicates that our models indeed capture users' interests from different perspectives; here with respect to emergent (latent) music genres and annotation taxonomy.

Figure 8a shows that a (small) disassortative mixing pattern exists between user popularity and focus for all our models. Users' latent tastes tend to disperse slightly as the number of their friends increases. We used Jensen-Shannon divergence (JS) to analyze the similarity between popular users (i.e., users with many social ties) and their neighbors. We found that as users popularity increases, so does topical divergence



(a) (b)

Fig. 8: (a) Average focus of users having k friends. (b) Average Jensen-Shannon divergence between all combinations of users having k friends and their friends.

with their ties. Figure 8b summarizes the results. The effect can be observed for all models, even though large fluctuations are apparent due to the small number of user pairs over which averaging is performed. This suggests that more and more diverse friendships are created with increasing user popularity. This phenomenon discloses the cognitive process of a user's friending behavior.

7. NETWORK RECONSTRUCTION & USER HOMOPHILY

Link prediction is important in social networks for understanding the mechanisms by which social networks form and evolve [Ge and Zhang 2012]. Most approaches thus far assume that a snapshot of the social network, with some links missing, is available. Ge and Zhang [2012] proposed a two-phase supervised method to address the problem of predicting the structure of a social network when only a small subgraph of the social network is known and multiple heterogeneous sources are available. A recent study [Leroy et al. 2010] has discussed the link prediction problem when the network is not fully observed. Mislove et al. [2010] explored the complementary question: can we predict topical similarity from the social network?

In our work, we evaluate the effectiveness of our approach with respect to the task of extracting the structure of the social network, i.e., all links at the same time. In this scenario, no prior friendship links are provided to our method. Since no friendship links are available, it is impossible to exploit the topological structure of the social network. Here, we focus on latent-based network reconstruction, where our objective is to reveal all links between pairs of users through their pair-wise similarity. The novelty of our approach comes from the fact that we combine topological structure with inferred latent user profiles, which are described as distributions over resources and their associated metadata, instead of actual content [Lipczak et al. 2012; Makrehchi 2011].

We examine the performance of SLIgHT (see Section 4) for each of our three tripartite graph generative models. We compute the Accuracy, Precision, Recall and F-measure of this approach while varying the number of hidden topics ($T = \{1, 10, 20, 50, 100\}$). The optimal threshold in each case is selected using Equation (6). Figure 9 shows the results. All three models are able to yield very high accuracy, however, their precision and recall are low for practical purposes. This result seems to contradict the hypothesis of user homophily in social networks [McPherson et al. 2001],

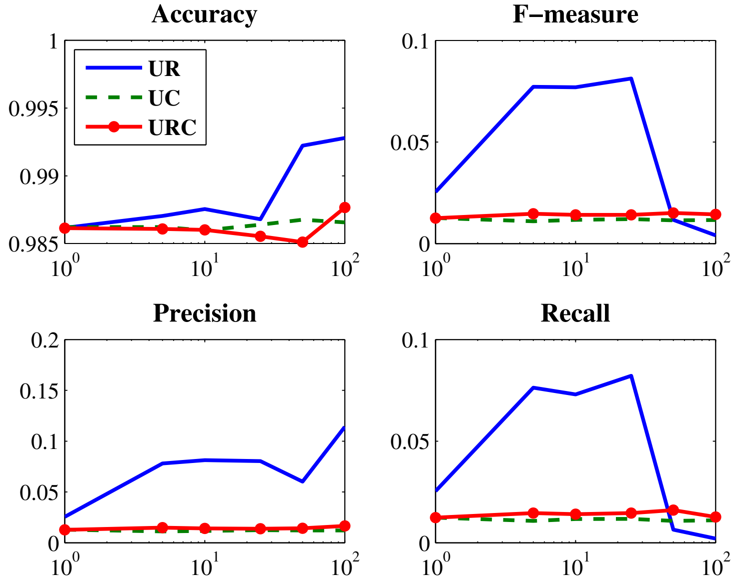


Fig. 9: Performance of SLIgHT. X-axis: number of hidden topics; Y-axis: Performance on test set.

since users' interests in terms of hidden topics are not accurate predictors of link formation. In fact, high accuracy values are observed due to correct classification of true negatives (absent of links).

Our results contradict those of Makrehchi [2011] for academic co-authorship networks. We explain this as a result of the very well defined structure and focused nature of co-authorship networks. Instead, online social networks encompass diverse user communities, which may or may not be related to each other. Katz score is used in this approach to calculate users proximity in the latent space defined by extracted hidden topics. Few heavily weighted paths in academic networks guarantee better results than many long (weak) paths in diverse, online social networks. Latent similarity with respect to artists yields better results than latent similarity with respect to tags. Similar musical preferences between users yield better predictive power with respect to link prediction in Last.fm as a result. Our URC model exhibits inferior performance than UR due to its attempt to capture similarity in terms of annotations as well. URC and UC are therefore comparable in performance with respect to network reconstruction.

In the following sections, we focus our analysis on UR, UC, and URC models for $T_{UR} = 20$, $T_{UC} = 20$, and $T_{URC} = 50$ hidden topics respectively. This selection is based on optimal values achieved by the three models with respect to F-measure (see Figure 9) in the network reconstruction task. Different datasets and different settings (e.g., number of hidden topics) may lead to different results than what we report in this work.

7.1. Users' Homophily

Next, we analyze in detail the similarity of users' topic distributions in relation to their number of common friends and their distance d along the social network. Intu-

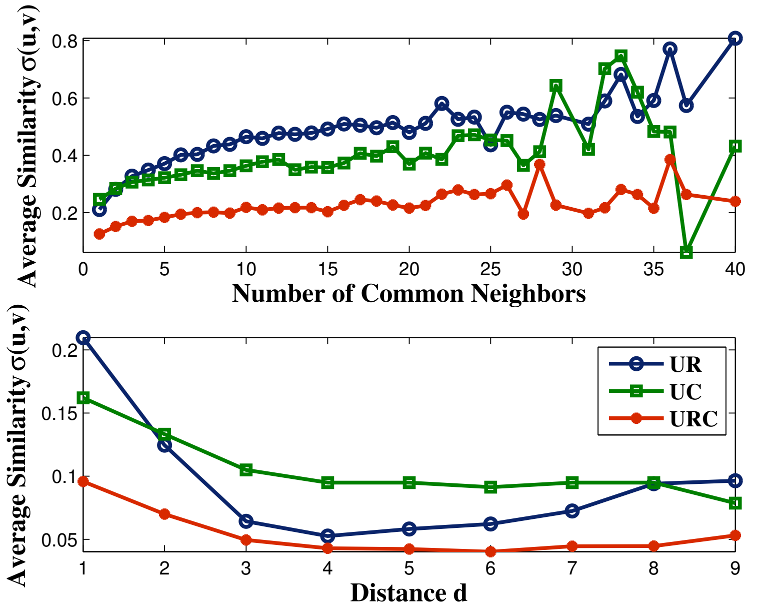


Fig. 10: Average similarity between latent topic vectors of Last.fm users as a function of (a) number of common neighbors and (b) distance d.

itively, the presence of a social tie indicates some degree of shared context between connected users, who are likely to have some interests in common [Schifanella et al. 2010]. Likewise, the existence of numerous common friends suggests sharing of common experiences through indirect interactions. Regardless of the mechanism driving this potential alignment, we measure this effect as a function of local structural properties. Figures 10a and 10b demonstrate these correlations respectively. The similarity score is calculated as cosine similarity between topic vectors from matrix Θ, using Equation (8). To compute averages for these quantities and exclude biases due to sampling, we performed an exhaustive investigation of the social network up to distance equal to the network diameter.

Figure 10a indicates strong alignment between users sharing numerous friends. Precisely, average similarity is large for large values of common friends, however, it drops as the number of common friends decreases. Large fluctuations in this case are visible for large number of common friends due to the small number of users over whom averages are computed. Average similarity under the URC model is relatively constant in this case, when the number of common neighbors is in the range between 1 and 25, even though a small increasing trend is visible. Hence, the existence of many common friends indicates interests commonality, which however may be distributed across different topics, for different subsets of common friends.

Similarly, Figure 10b suggests that a certain degree of alignment between neighbors in the social network is in fact existent. While average similarity is quite large for neighbors ($d = 1$), it drops rapidly as d increases and is close to zero for $d \geq 3$. Our observation corroborates the results presented by Schifanella et al. [2010], suggesting that the alignment of users' interests must be a local effect. Average similarity under the UC model is relatively constant for $d \in [3, 8]$, indicating common tag usage by many Last.fm users who have not established friendship relationships with each other. This fact explains why UC and URC perform worse than UR in the network reconstruction task (see Figure 9).

8. PREDICTION OF SOCIAL TIES

In this section, we test the effectiveness of our four classification schemes. We refer to “Latent Topics & Common Neighbors Scheme” as Scheme A, to “Latent Topics & Shortest Distance Scheme” as Scheme B, and to “Latent Topics Classification Scheme” as Scheme C. Finally, we refer to “Ensemble Classification Scheme” as Scheme D. Figure 11 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 “Ensemble Classification Scheme”' (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 extent, the aggregation methodology is non-linear to the preprocessing of true positives and true negatives samples, resulting in information loss in exchange of scalability gains.

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 not as high when compared to the rest of the schemes, it’s 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 seems to be increasing or stay constant for dataset size up to 50%, after which point over-fitting causes degradation in performance. 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 as well as user-generated content type in this context have to be considered when making this selection.

Support Vector Machines have to achieve a trade-off between maximizing the margin and minimizing the empirical error, which leads to classifying every sample to the dominant class (negative in our case) under high class imbalance or when data are non-separable, if the misclassification penalty is adequately small. This results in no (or minuscule) classification errors on the negative instances, but high errors on the positive instances, which even though are quite sparse, are also the most vital in social tie prediction. A classifier that classifies everything as negative may be extremely accurate but it will not have any practical use as it will never identify the positive instances correctly [Ertekin et al. 2007]. Due to social networks sparsity, we expect most test links to belong to the negative class (absence of link). We address this problem here by examining the Precision and Recall that our various schemes achieve when calculated separately for the positive and negative classes.

Figure 12 shows the results. Intuitively, true negatives are easier to classify correctly under most models, in most cases. Overall, we observe a degradation in performance with respect to true positives (which are harder to predict) due to over-fitting and noisy observations as the training dataset size increases. Nevertheless, all of our schemes yield reasonable results for practical purposes, for reasonably small training datasets (less than 20% of complete dataset in all cases). Based on the analysis presented above we observe that hidden topics proximity alone is not sufficient to accurately predict

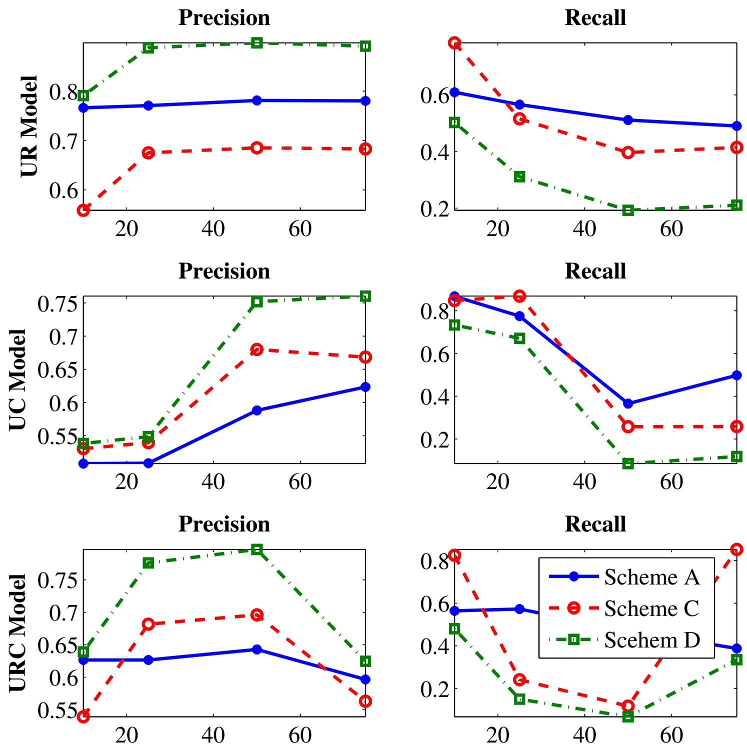


Fig. 11: Precision and Recall of Latent Semantics Classification Schemes as a function of training data size. X-axis: Training set size as percentage of complete dataset; Y-axis: Precision/Recall.

social ties. However, our work demonstrates that the addition of local network features to latent semantics greatly improves performance, often by a considerable margin.

8.1. Comparison with other methods

In this section, we compare our schemes with two tag-based similarity metrics, which have shown superior performance in the content-based network reconstruction task [Schifanella et al. 2010]:

(1) Cosine Similarity (CS). The normalized cosine similarity between two users u and v can be calculated as follows: $CS(u,v) = \frac{\sum_t f_u(t) f_v(t)}{\sqrt{\sum_t f_u(t)^2 \sum_t f_v(t)^2}}$, where $f_u(t)$ denotes the number of times user u has used tag t.

(2) Maximal Information Path: Similarity metric that computes semantic relatedness of terms in non-hierarchical triple representation [Schifanella et al. 2010].

We present results in the form of the area under the receiver-operating characteristic curve (AUC). AUC quantifies prediction accuracy and tests how much better a classifier is than pure chance, while at the same time measuring its overall ability

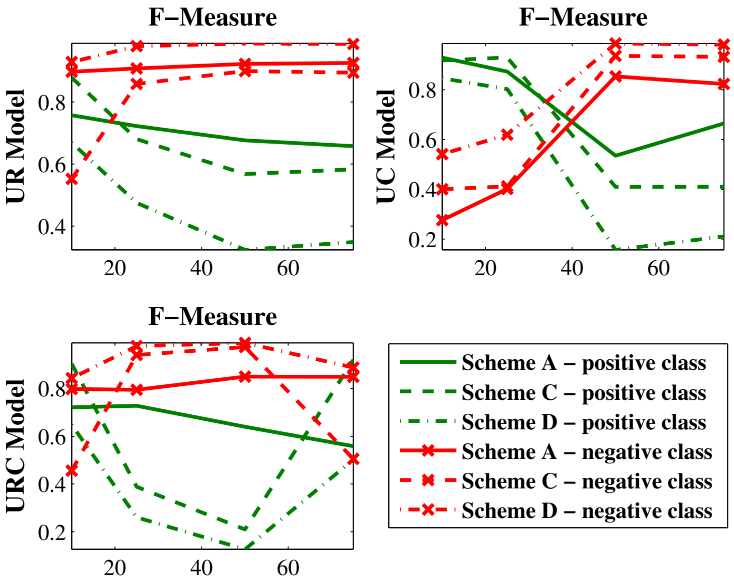


Fig. 12: F-measure (calculated for positive & negative classes separately) achieved by Latent Semantics Classification Schemes as a function of training data size. X-axis: Training set size as percentage of complete dataset; Y-axis: F-measure.

to rank all missing connections (positive class), which are the hardest to predict, over nonexistent ones (negative class). AUC evaluates classification performance across the entire range of decision thresholds, providing a good performance overview when the operating condition for the classifier is unknown or the classifier is expected to be used in situations with significantly different class distributions.

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. 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. Particularly, 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. Our “Ensemble Classification Scheme” (Scheme D) produces only class labels without assigning score values, hence we exclude it from our comparison. This leaves us with two Schemes, A and C.

The choice to select pairs of users randomly stems from the size of our dataset, which makes an exhaustive comparative evaluation infeasible. 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. Note that this strategy may bias the evaluation in favor of the baselines, which have a complete view of the dataset for their similarity calculations. Therefore, our evaluation is a conservative choice in that it does not unfairly help our proposed schemes (in fact there might be a bias against them).

For consistency across our experiments, we focus our analysis on UR, UC and URC models for $T_{UR} = 20$, $T_{UC} = 20$, and $T_{URC} = 50$ hidden topics respectively. This setup limits our observations to three settings only. However, it would be tedious and diffi-

Table III: Area under the ROC curve comparison for 10%, 25%, 50%, and 75% of edges observed

Model	**Scheme**	**% of observed edges**			
		10%	25%	50%	75%
UR	Scheme A	0.5624	0.6569	0.8663	0.8949
	Scheme C	0.5454	0.6005	0.6418	0.7342
UC	Scheme A	0.5514	0.7129	0.7993	0.8511
	Scheme C	0.5500	0.6225	0.6429	0.7417
URC	Scheme A	0.6515	0.7007	0.7967	0.8540
	Scheme C	0.6491	0.5485	0.6357	0.7654
Baselines	MIP	0.6256			
	CS	0.6087			

cult to compare all our models for all settings respectively. Further, different datasets may result in different optimal models. Table III shows the results. The observation of AUC values further validates that our classification schemes act as proper ranking functions for all three models. Scheme A, which combines latent topics with local structure (number of common neighbors), performs better than Scheme C, which only considers latent similarity. Further, as the fraction of observed edges increases, the classification accuracy of our schemes improves significantly. When 20% or more of the original dataset is provided for training, our schemes outperform the baselines, often by a considerable factor.

9. RELATED WORK

Probabilistic models have been successfully used in discovering the hidden topics that were responsible for generating a collection of documents [Blei et al. 2003]. Our model is an adaptation of the author-topic model proposed by Rosen-Zvi et al. [2010]. The objective of their work was to provide a generative process for document creation, capable of recovering hidden topics in a document corpus. We are extending their model to resources of any type (not just documents) and annotations (instead of words). The social process of annotation generation is unknown. It is not intuitive that such framework would perform as well in this context.

Social tagging systems have been well studied, leading to a vast literature around this area. Gupta et al. [2010] summarized different techniques employed to study various aspects of tagging. Halpin et al. [2007] studied the basic dynamics behind tagging in the social bookmarking site del.icio.us[4] and proposed a collaborative tagging model based on preferential attachment and informational value. We instead take a probabilistic, generative approach that accurately models collaborative annotation in online social media.

Bundschus et al. [2009] proposed a model, which does not correctly simulate the real social annotation process because users are modeled as creators of content words instead of tags. Lu et al. [2010] proposed a model that overcomes the limitations of previous models by representing all related entities (users, documents, words and tags) and latent variables (topics, user perspectives) in a unified model. Their model exhibits high complexity due to the numerous variables that have to be estimated, and does not sufficiently capture users' interests, as in our case. Harvey et al. [2011] proposed to use hidden topic models to improve social bookmark search results. Hariri et al.

[4] https://delicious.com/

[2012] proposed a context-aware music recommendation system, which leverages top frequent tags for songs from social tagging Web sites, using Latent Dirichlet Allocation to determine a set of latent topics for each song. Our models can be effectively used to recommend not only resources, but also tags and users at the same time.

Lin et al. [2012] explored tag growth and users' activities dynamics in social media using a model that resembles ours. Their approach differs from ours in that they model posts that contain resources and tags, whereas we are modeling direct annotation of resources. Liu et al. [2012] proposed a framework to combine the tasks of user preference discovery and document topic mining through modeling of user-document interactions. In both works, there is only one "tagger" per document, whereas our model captures the social aspect of tagging, allowing a mixture of users to collaboratively contribute in the annotation process. Therefore, our approach is more general and the problem we study more difficult. Long et al. [2006] proposed a general model to find hidden structures (local clusters and global community structures) from a k-partite graph. By introducing hidden nodes into the original k-partite graph, they construct a relation summary network to approximate the original k-partite graph under a broad range of distortion measures. We instead are focusing on the actual generative process that drives the original tripartite graph creation.

Relational topic model was introduced to model links between documents as binary random variable conditioned on their contents [Chang and Blei 2009]. Topic-link model [Liu et al. 2009] performed topic modeling and author community discovery in a unified framework but did not provide reasonable results in the task of link prediction. Pennacchiotti and Gurumurthy [2011] applied LDA for social link recommendation, modeling social media users' streams as documents, represented by words that they emit in social media. Krestel et al. [2009] applied LDA for tag recommendation. Even though our approaches are similar, we utilize resources and annotations as descriptors of user interests and we propose three generative models that capture the essence of tripartite graph formation in social networks. We further demonstrate that our approach yields high precision and recall in the social link recommendation task.

The problem of link prediction for social networks has been well studied in numerous domains and contexts. Lu and Zhou [2011] explored several network proximity metrics for social link prediction, demonstrating that important information can be mined from the graph alone. Schifanella et al. [2010] utilized vocabulary overlap between users as indicator of user connectivity in Flickr. While we are adopting this hypothesis, we instead propose a generative process to model content annotation by users. Moreover, we consider this process in conjunction with local network structure. Taskar et al. [2003] proposed a relational Markov network framework to define a joint probabilistic model over the entire graph-entity attributes and links, assuming a Markov dependency (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.

Backstrom and Leskovec [2011] predicted and recommended links in social networks using random walks. Unlike ours, their approach depends on knowing almost all links along with a set of source and candidate nodes, and only needs to predict few new links. Sadilek et al. [2012] predicted friendship links in Twitter based on one input feature, assuming mutual independence between the observed and hidden variables. Their model exhibits high complexity due to the vast number of hidden nodes it includes (one for each possible link). Both approaches only consider undirected graphs. Instead, we test the effectiveness of our approach in a directed social network, which captures more realistically the asymmetric relationships between users.

Recently, the problem of link prediction in heterogeneous networks has been studied. Sun et al. [2011] proposed PathSim to measure the similarity among same type objects in heterogeneous networks based on symmetric meta paths (sequence of relations between different object types). However, in online social networks many valuable paths are asymmetric and the relatedness of different-typed objects is also meaningful. Hence, PathSim is not suitable in this context. Further, there is no standard procedure to follow in order to explicitly specify path combinations to define meta paths. Typical users do not necessarily posses the amount of domain knowledge required to define meta paths. Choosing the best path by experimentation or learning it from training examples leads to a state space explosion, rendering this approach impractical. Davis et al. [2011] proposed a neighborhood multi-relational link prediction approach based on triad census, trivially extended to heterogeneous networks. Its weighted version is equivalent to weighted common neighbors. Prediction scores are calculated individually for each link type of interest, ignoring latent influence due to meta paths. Instead, our approach combines structural features with latent user interests, while at the same time provides a generative model of the heterogeneous network formation and evolution.

Latent feature based models [Hoff 2009; Menon and Elkan 2011] consider link prediction 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. Our model, as an adaptation of the author-topic model, is closely related to methods based on matrix factorization [Rosen-Zvi et al. 2010]. For applications where models with n-ary relations with $n > 3$ need to be considered, tensor factorization techniques are required. Kolda and Bader [2009] provide a recent overview of leading approaches. Unfortunately, the straightforward application of higher-order tensor models becomes problematic due to computational requirements and data sparsity.

Dietz [2009] proposed a generative model that learns shared tastes of users from network structure and user playlists. Even though our approach is similar in spirit, our user modeling radically differs. Perhaps the work closest to ours is that of Parimi and Caragea [2011]. 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. Further, in their small-scaled experiments, they only considered ROC-AUC analysis. We, in contract, address scalability issues, considering thousands of users who may be arbitrarily connected, resulting in million potential friendships. Last but not least, we show that our approach effectively addresses the high class imbalance problem due to data sparsity.

10. CONCLUSIONS

In this article, we presented three generative probabilistic models of online social tagging systems as a principled way of reducing the dimensionality of such data, capturing at the same time the dynamics of collaborative annotation process. Our models represent users' interests in a latent space over resources and rich metadata describing them. Even though our 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 our generative probabilistic models can be used to learn users' tastes and to effectively reconstruct the network of ties or predict future social links when some prior evidence is provided. In particular, we showed how to exploit latent user interests in conjunction with structural features to significantly improve social link prediction in the online music social media site Last.fm. We showed that similarity of interests alone does not trigger the creation of a social link. Instead, we showed how to achieve high prediction performance using four classifiers, which jointly exploit users' interests similarity and their local network proximity. We plan to further validate our results by examining dynamic social networks. Taking into account temporality, we will be able to better understand if the combination of taste and local network similarity indeed drives tie formation or if conversely, tie formation results in taste alignment and local network densification. While most link prediction methods suffer from the high class imbalance problem resulting in low precision and/or recall solutions, our proposed methods achieve high precision and recall for highly imbalanced classes.

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 the quality of predictions. In our future work, we plan to address the challenge of combining multiple heterogeneous sources of information 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. Besides link recommendation and prediction, our methods can be easily adapted to facilitate analysis of trending topics and users' latent interests, resource and tag recommendations and categorization, classification and filtering of online information.

ACKNOWLEDGMENT

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

REFERENCES

Lars Backstrom and Jure Leskovec. 2011. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the fourth ACM international conference on Web search and data mining (WSDM '11)*. ACM, New York, NY, USA, 635–644.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.* 3 (March 2003), 993–1022.

Markus Bundschus, Shipeng Yu, Volker Tresp, Achim Rettinger, Mathaeus Dejori, and Hans-Peter Kriegel. 2009. Hierarchical Bayesian Models for Collaborative Tagging Systems. In *Proceedings of the 2009 Ninth IEEE International Conference on Data Mining (ICDM '09)*. IEEE Computer Society, Washington, DC, USA, 728–733.

Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. 2011. 2nd Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011). In *Proceedings of the 5th ACM conference on Recommender systems (RecSys 2011)*. ACM, New York, NY, USA.

Jonathan Chang and David Blei. 2009. Relational topic models for document networks. In *In Proc. of Conf. on AI and Statistics (AISTATS*.

Nello Cristianini and John Shawe-Taylor. 2010. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press. I–XIII, 1–189 pages.

Darcy Davis, Ryan Lichtenwalter, and Nitesh V. Chawla. 2011. Multi-relational Link Prediction in Heterogeneous Information Networks. In *Proceedings of the 2011 International Conference on Advances in Social Networks Analysis and Mining (ASONAM '11)*. IEEE Computer Society, Washington, DC, USA, 281–288.

Laura Dietz. 2009. Modeling Shared Tastes in Online Communities. In *NIPS Workshop on Applications for Topic Models: Text and Beyond*.

Seyda Ertekin, Jian Huang, Leon Bottou, and Lee Giles. 2007. Learning on the border: active learning in imbalanced data classification. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management (CIKM '07)*. ACM, New York, NY, USA, 127–136.

Liang Ge and Aidong Zhang. 2012. Pseudo Cold Start Link Prediction with Multiple Sources in Social Networks. In *SDM*. SIAM / Omnipress, 768–779.

Scott Golder and Bernardo A. Huberman. 2006. The Structure of Collaborative Tagging Systems. *Journal of Information Science* 32, 2 (April 2006), 198–208.

Mark Granovetter. 1983. The Strength of Weak Ties: A Network Theory Revisited. *Sociological Theory* 1 (1983), 201–233.

Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America* 101, Suppl 1 (2004), 5228–5235.

Manish Gupta, Rui Li, Zhijun Yin, and Jiawei Han. 2010. Survey on social tagging techniques. *SIGKDD Explor. Newsl.* 12, 1 (Nov. 2010), 58–72.

Harry Halpin, Valentin Robu, and Hana Shepherd. 2007. The complex dynamics of collaborative tagging. In *Proceedings of the 16th international conference on World Wide Web (WWW '07)*. ACM, New York, NY, USA, 211–220.

Negar Hariri, Bamshad Mobasher, and Robin Burke. 2012. Context-aware music recommendation based on latenttopic sequential patterns. In *Proceedings of the sixth ACM conference on Recommender systems (RecSys '12)*. ACM, New York, NY, USA, 131–138.

Morgan Harvey, Ian Ruthven, and Mark J. Carman. 2011. 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)*. ACM, New York, NY, USA, 485–494.

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

Donald B. Johnson. 1977. Efficient Algorithms for Shortest Paths in Sparse Networks. *J. ACM* 24, 1 (Jan. 1977), 1–13.

Leo Katz. 1953. A new status index derived from sociometric analysis. *Psychometrika* 18 (1953), 39–43. Issue 1. 10.1007/BF02289026.

S. Sathiya Keerthi, Olivier Chapelle, and Dennis DeCoste. 2006. Building Support Vector Machines with Reduced Classifier Complexity. *J. Mach. Learn. Res.* 7 (Dec. 2006), 1493–1515.

Tamara G. Kolda and Brett W. Bader. 2009. Tensor Decompositions and Applications. *SIAM Rev.* 51, 3 (Aug. 2009), 455–500.

Ralf Krestel, Peter Fankhauser, and Wolfgang Nejdl. 2009. Latent dirichlet allocation for tag recommendation. In *Proceedings of the third ACM conference on Recommender systems (RecSys '09)*. ACM, New York, NY, USA, 61–68.

Kristina Lerman and Anon Plangprasopchok. 2009. *Handbook of Research on Web 2.0, 3.0, and X.0: Technologies, Business, and Social Applications*. IGI Global, Chapter Leveraging User-specified Metadata to Personalize Image Search.

Vincent Leroy, B. Barla Cambazoglu, and Francesco Bonchi. 2010. Cold start link prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '10)*. ACM, New York, NY, USA, 393–402.

Nan Lin, Daifeng Li, Ying Ding, Bing He, Zheng Qin, Jie Tang, Juanzi Li, and Tianxi Dong. 2012. The dynamic features of delicious, flickr, and YouTube. *J. Am. Soc. Inf. Sci. Technol.* 63, 1 (Jan. 2012), 139–162.

Marek Lipczak, Borkur Sigurbjornsson, and Alejandro Jaimes. 2012. Understanding and leveraging tag-based relations in on-line social networks. In *Proceedings of the 23rd ACM conference on Hypertext and social media (HT '12)*. ACM, New York, NY, USA, 229–238.

Lu Liu, Feida Zhu, Lei Zhang, and Shiqiang Yang. 2012. A probabilistic graphical model for topic and preference discovery on social media. *Neurocomputing* 95 (Oct. 2012), 78–88.

Yan Liu, Alexandru Niculescu-Mizil, and Wojciech Gryc. 2009. Topic-link LDA: joint models of topic and author community. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML '09)*. ACM, New York, NY, USA, 665–672.

Zhiyuan Liu, Yuzhou Zhang, Edward Y. Chang, and Maosong Sun. 2011. PLDA+: Parallel latent dirichlet allocation with data placement and pipeline processing. *ACM Trans. Intell. Syst. Technol.* 2, 3, Article 26 (May 2011), 18 pages.

Bo Long, Xiaoyun Wu, Zhongfei (Mark) Zhang, and Philip S. Yu. 2006. Unsupervised learning on k-partite graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '06)*. ACM, New York, NY, USA, 317–326.

Caimei Lu, Xiaohua Hu, Xin Chen, Jung-Ran Park, TingTing He, and Zhoujun Li. 2010. 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)*. ACM, New York, NY, USA, 683–692.

Linyuan Lu and Tao Zhou. 2011. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications* 390, 6 (2011), 1150–1170.

Masoud Makrehchi. 2011. Social link recommendation by learning hidden topics. In *Proceedings of the fifth ACM conference on Recommender systems (RecSys '11)*. ACM, New York, NY, USA, 189–196.

Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology* 27, 1 (2001), 415–444.

Aditya Krishna Menon and Charles Elkan. 2011. 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)*. Springer-Verlag, Berlin, Heidelberg, 437–452.

Alan Mislove, Bimal Viswanath, Krishna P. Gummadi, and Peter Druschel. 2010. You are who you know: inferring user profiles in online social networks. In *Proceedings of the third ACM international conference on Web search and data mining (WSDM '10)*. ACM, New York, NY, USA, 251–260.

Rohit Parimi and Doina Caragea. 2011. 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)*. Springer-Verlag, Berlin, Heidelberg, 75–86.

Marco Pennacchiotti and Siva Gurumurthy. 2011. Investigating topic models for social media user recommendation. In *Proceedings of the 20th international conference companion on World wide web (WWW '11)*. ACM, New York, NY, USA, 101–102.

John C. Platt. 1999. Advances in kernel methods. MIT Press, Cambridge, MA, USA, Chapter Fast training of support vector machines using sequential minimal optimization, 185–208.

Michal Rosen-Zvi, Chaitanya Chemudugunta, Thomas Griffiths, Padhraic Smyth, and Mark Steyvers. 2010. Learning author-topic models from text corpora. *ACM Trans. Inf. Syst.* 28, 1, Article 4 (Jan. 2010), 38 pages.

Adam Sadilek, Henry Kautz, and Jeffrey P. Bigham. 2012. Finding your friends and following them to where you are. In *Proceedings of the fifth ACM international conference on Web search and data mining (WSDM '12)*. ACM, New York, NY, USA, 723–732.

Rossano Schifanella, Alain Barrat, Ciro Cattuto, Benjamin Markines, and Filippo Menczer. 2010. 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)*. ACM, New York, NY, USA, 271–280.

Shai Shalev-Shwartz and Nathan Srebro. 2008. SVM optimization: inverse dependence on training set size. In *Proceedings of the 25th international conference on Machine learning (ICML '08)*. ACM, New York, NY, USA, 928–935.

Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S. Yu, and Tianyi Wu. 2011. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. In *In VLDB 11*.

Ben Taskar, Ming fai Wong, Pieter Abbeel, and Daphne Koller. 2003. Link Prediction in Relational Data. In *in Neural Information Processing Systems*.

Ivor W. Tsang, James T. Kwok, and Pak-Ming Cheung. 2005. Core Vector Machines: Fast SVM Training on Very Large Data Sets. *J. Mach. Learn. Res.* 6 (Dec. 2005), 363–392.

S. Wasserman and K. Faust. 1994. *Social network analysis: Methods and applications*. Cambridge Univ Press.

Rongjing Xiang, Jennifer Neville, and Monica Rogati. 2010. Modeling relationship strength in online social networks. In *Proceedings of the 19th international conference on World wide web (WWW '10)*. ACM, New York, NY, USA, 981–990.

Received Month Year; revised Month Year; accepted Month Year