Social Link Recommendation by Learning Hidden Topics

Masoud Makrehchi
Thomson Reuters
610 Opperman Dr, Saint Paul, MN 55123, USA
masoud.makrehchi@thomsonreuters.com

ABSTRACT

In this paper, a new approach to predicting the structure of a social network without any prior knowledge from the social links is proposed. In absence of links among nodes, we assume there are other information resources associated with the nodes which are called node profiles. The task of link prediction and recommendation from text data is to learn similarities between the nodes and then translate pair-wise similarities into social links. In other words, the process is to convert a similarity matrix into an adjacency matrix. In this paper, an alternative approach is proposed. First, hidden topics of node profiles are learned using Latent Dirichlet Allocation. Then, by mapping node-topic and topic-topic relations, a new structure called semi-bipartite graph is generated which is slightly different from regular bipartite graph. Finally, by applying topological metrics such as Katz and short path scores to the new structure, we are able to rank and recommend relevant links to each node. The proposed technique is applied to several co-authorship networks. While most link prediction methods are low precision solutions, the proposed method performs effectively and offers high precision. **Categories and Subject Descriptors:** I.2.6 [Artificial Intelligence]: Learning **General Terms:** Algorithms. **Keywords:** link recommendation, social network, Latent Dirichlet Allocation.

1. INTRODUCTION

Social network is defined as a map of relationship between individuals. Although social networks are usually employed as descriptive models, recent research on machine learning and data mining have introduced methods and algorithms to build statistical models of network data including social networks, web-page networks, email tracks, and citation networks [6, 10]. These models can be obtained either directly from data using information extraction algorithms, which are applied mostly to relational database and semi-structured text, or indirectly from unstructured text data using text mining techniques. In the first case, the concept of acquaintanceship (or link between two nodes or interacting units) can be extracted from information in the data itself. In this case, the extracted network is descriptive rather than predictive, and is primarily to visualize and analyze the extracted network [3, 9]. For another example, in [15], social networks are generated by mining knowledge-sharing sites to support generating a marketing plan. In the knowledge-sharing networks, customers share their opinions with others. The social network is generated by a probabilistic model of the network.

Data mining has been also used for mining social networks. For instance, in [11], using the influence diffusion model, the social networks in a message board are extracted. The model is simply based on the frequency of terms which are propagated between two individuals. The extracted social networks are categorized to study some social and psychological issues such as interactivity among members. Another approach to extracting social networks is employing usage and log data instead of textual content. One application is exploring the social networks in instant messaging systems to study the network related issues such as network traffic [14].

Acquaintanceships and ties in a social network might also be extracted using citation, web link, and co-authorship. In [7], social network of software reverse engineering community is extracted from co-authorship graph. It shows that the community behaves like a small world. This result confirms Watts and Strogatz claim [4], in which social networks are assumed to be small world.

In the second approach, which is more predictive, acquaintanceship is translated into similarity or proximity of two nodes. This pair-wise similarity can be extracted from any textual resources of the nodes. A textual resource can be any article, paper, news, resume, personal web page, CV, and so on [13].

Text-based similarity is not the only way to approximately explore the acquaintanceship between individuals. In [12], the number of times that two individuals appear in a web page is considered as a degree of acquaintanceship to build a social link. The method has been applied to extract social relationships of conference participants. One drawback of querying names in a search engine is the problem of distinct identities. Since for social network extraction, we need to assign a unique identity to each node, using a search engine, many individuals may have same identities and a person may be known by more than one name. Using FOAF per-

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.
RecSys'11, October 23–27, 2011, Chicago, Illinois, USA.
Copyright 2011 ACM 978-1-4503-0683-6/11/10 ...$10.00.

son metadata, this problem can be resolved because each member is associated with a unique URI.

This paper proposes a new approach to learning social networks from text. The proposed approach requires a set of documents associated with members of the community. Using vector space model document representation, each person is described by a set of terms (single unique words).

Associating the people in a community with the terms in a dictionary, a new structure called *node-term matrix* is introduced. Similar to document-term matrix, the new model suffers from high dimensionality of the feature space. By a set of preprocessing tasks such as stemming, stopword removal, and document frequency thresholding, the number of terms is dramatically reduced. In addition to vector space model representation, Latent Semantic Indexing is used for representing nodes and also dimensionality reduction. In order to assign most relevant topics to a predicted social links between nodes, Latent Dirichlet Allocation model is employed. Using this model, every node is represented by a set of hidden topics. Then, by mapping node-topic and topic-topic relations, a new structure called semi-bipartite graph is generated which is slightly different from regular bipartite graph. Finally, by applying link prediction metrics such as Katz and short path scores to the new structure, we are able to rank and recommend relevant links to each node.

The paper consists of six sections. After the introduction, the data set used in this paper is detailed. The problem of learning social links from text data is briefly introduced in Section 3. In Section 4, similarity-based link prediction is discussed. the proposed approach, topic-based link prediction, is detailed in section 5. The experimental result and discussion are presented in section 5, followed by concluding remarks in the section 6.

2. DATA SET

Bibliographic information of papers published in 20 scientific domains such as Acoustics, Dermatology, Microbiology, Statistics, and Zoology are collected from the Web (see Table 1). Each data item contains title, list of authors, and abstract. In each domain, there is an explicit co-authorship network which can be derived directly from the author list of the papers. The goal is to predict the co-authorship links by translating similarity between abstracts into acquaintanceship (co-authorship) between the authors. Using predicted links, we are able to generate an implicit co-authorship network. By comparing the generated, implicit network with the derived, explicit one, we can evaluate the proposed method.

In order to perform the experiments, the networks derived from authorship information of the data, are employed as test data set. Since the test data has to be completely isolated from training data, first, only authors (author set $\mathbb{A}$) who have both sole-authored and co-authored papers are selected. Then, the documents associated with the list $\mathbb{A}$, which is called $\mathbf{d}$, are selected and the remaining data are removed. The relationship between the set $\mathbf{d}$ and $\mathbb{A}$ is a many-to-many relationship. The new set of documents $\mathbf{d}$ is divided into two subsets $\mathbf{d}_{single}$ and $\mathbf{d}_{multi}$ such that $\mathbf{d} = \mathbf{d}_{single} \cup \mathbf{d}_{multi}$ where $\mathbf{d}_{single}$ is the set of sole-authored papers and $\mathbf{d}_{multi}$ is the set of co-authored papers. In the next step, the authorship information of the papers in $\mathbf{d}_{multi}$ is retrieved. This information which is the test data is used to extract explicit co-authorship links. On the other hand, $\mathbf{d}_{single}$ is employed to build the training data. Every document in $\mathbf{d}_{single}$ is associated with an author. Every single abstract is retrieved and assigned to its corresponding author. In the case an author having more than one paper, abstracts of papers are merged to build one single document. Using Bag-Of-Words (BOW) representation, every author (node in social network terminology) is represented by a set of words. The goal is to predict implicit co-authorship network using authors' abstracts and compare with the true links extracted from co-authorship data ($\mathbf{d}_{multi}$).

Table 1: The list of co-authorship networks employed in this paper (n: number of nodes; r: number of links; and S: network sparsity).

id	network	n	r	S_p
1	architecture	245	294	0.9902
2	biochemical research methods	366	953	0.9857
3	biology	382	635	0.9913
4	biomedical	270	569	0.9843
5	cardiac	304	891	0.9807
6	dietetics	314	749	0.9848
7	hardware	245	294	0.9902
8	hematology	468	1457	0.9867
9	mathematical	304	478	0.9896
10	medical informatics	217	515	0.9780
11	medicinal	208	903	0.9581
12	neuroimaging	202	633	0.9688
13	nutrition	314	749	0.9848
14	ophthalmology	331	373	0.9932
15	pathology	305	698	0.9849
16	peripheral vascular disease	284	1057	0.9737
17	physics	245	315	0.9895
18	probability	293	246	0.9942
19	surgery	391	1074	0.9859
20	zoology	308	648	0.9863

3. LEARNING SOCIAL LINKS FROM TEXT DATA

A social network is considered as a descriptive framework to study and analyze social relations and behaviors. In this paper, social networks are viewed as predictive rather than descriptive models. The problem is, in the lack of an explicit social network, how we can, approximately, predict and learn the implicit network while knowing only the similarity between nodes.

Let $\mathbb{A} = \{a_1, a_2, ..., a_n\}$ be a finite set of n nodes in the community. The nodes are connected to each other through a finite set of relations or links to form a social network. For example, $\mathbb{A}$ can be a group of authors who may collaborate and publish articles with each other. Since we assume the social relations to be symmetric, there are $N = n(n-1)/2$ possible links among the nodes. Practically, the set of existing relations is very small subset of the possible relations. This fact addresses an important characteristic of social network which is called social network sparsity S_p:

$$S_p = 1 - \frac{2r}{n(n-1)} \tag{1}$$

where r is the number of links in the network. Social networks are usually very sparse graphs. High sparsity can create isolated sub-networks or even isolated nodes. On the other hand, low sparsity increases the density of the network, in which every node is connected to the others.

Let's $\mathbf{d} = \{d_i | 1 \leq i \leq n\}$ be the corresponding set of documents associated with the set of nodes $\mathbb{A}$. The problem of learning social links between the nodes in $\mathbb{A}$ is to predict entries of the following adjacency matrix using documents in $\mathbf{d}$:

$$\mathbf{E} = \left(\begin{array}{cccc} 0 & e(1,2) & \cdots & e(1,n) \\ e(2,1) & 0 & \cdots & e(2,n) \\ \cdots & & & \\ e(n,1) & e(n,2) & \cdots & 0 \end{array} \right). \qquad (2)$$

where $e(i,j) =< a_i, a_j >$ is the edge between nodes i and j and $e(j,i) \in \{0,1\}$.

In most link prediction works, rarely a new node is introduced to the network. A network at time t is given, and we are interested in predicting the structure of the network at time $t+1$. In link prediction, the probability of a new link between two existing nodes is learned from their links and those of others. However, in learning social links, given a set of nodes, either some links among the nodes are already known or all links are missing. The former introduces a supervised learning social links problem while the latter is about an unsupervised problem. The proposed method in this paper is an unsupervised approach to predicting social links.

The idea is to build a social network from text documents through extracting semantic similarities between nodes which are associated with documents. It is based on translating the problem of finding relations between people into estimating similarities between them. In order to find the similarity, each node is represented by a set of relevant terms extracted from its corresponding document(s) such as personal home pages, papers or resumes.

In addition to information extraction techniques, natural language processing and text mining techniques may provide efficient frameworks to extract information about people. By employing text mining tasks and using a corpora, the social network generation algorithm learns non-trivial patterns, similarities and associations among nodes.

Let us suppose C be a community including a finite number (n) of nodes. Every node may be related to some others based on common interests and similarities. In order to extract the relations and build the graph of relationship between the nodes, we must model the similarities.

Modeling the common interests and familiarities is a subjective process, because it is about capturing some knowledge answering the question what can relate two nodes to each other in a community? It requires learning about similarities between the nodes through modeling the nodes in a unified framework. In other words, it is all about describing two persons with the same dictionary.

Three essential elements of the similarity-based approach to automatic social network generation are *(i)* information resources outlining the interests of the nodes; *(ii)* a measure for extracting pair-wise similarities between nodes and build similarity matrix; and *(iii)* a method to convert the similarity matrix into a binary adjacency matrix and visualize the resulting graph.

4. LEARNING PAIR-WISE SIMILARITY

Representing each node with a feature vector, the whole community can be represented by a structure called node-term matrix $\mathbb{A}$ in which each row is associated with a node and a column associated with a word in the dictionary. Similar to document-term matrix in text mining tasks, one major problem with node-term matrix is its high dimensionality. To deal with the high dimensionality of $\mathbb{A}$, for instance, in [13], keywords are extracted from the information resources such as web pages resulting a set of keywords representing every node. In the next step Jaccard similarity measure is applied to extracted keywords to derive relations between the nodes.

The problem with keyword extraction is that it is subjective, difficult, and less precise. One alternative approach to dealing with the problem of high dimensional space, is to reduce the number of terms with multiple preprocessing tasks and a low rank approximation method such as Latent Semantic Indexing (LSI). The preprocessing tasks, including stopword reduction, stemming, and document frequency threshold, can remove almost 80% of redundant, non-informative, and non-relevant terms, as well as spelling errors and noncontributing terms to the meaning of the document.

Associating nodes in a social graph with a set of terms, LSI can be employed to extract semantic relations between the nodes and then we can estimate the pair-wise similarity matrix. By applying a threshold to the matrix, a set of candidate friends for each node is recommended.

LSI has been inspired by using Singular Value Decomposition (SVD) to capture major associative patterns in the data. LSI not only reflects the significant associations, which can be interpreted as semantic relations, but also ignores the weak influences. What we achieve by applying LSI is effective dimensionality reduction and at the same time, extracting more relevant features.

From [2, 8], a rectangular matrix, such as the node-term matrix $\mathbf{A}$, can be decomposed into the product of three matrices including two orthogonal matrices ($\mathbf{U}$ and $\mathbf{V}$) and a diagonal ones ($\mathbf{\Sigma}$) as follows;

$$\mathbf{A} = \mathbf{U} \times \mathbf{\Sigma} \times \mathbf{V^T} \qquad (3)$$

where $\mathbf{\Sigma}$ is a full rank diagonal matrix. Sorting the singular values in $\mathbf{\Sigma}$, q largest values are kept and the rest are set to zero. The method is called truncated SVD for dimensionality reduction. The result is a rank-q model of the singular values matrix. By deleting low rank columns of $\mathbf{U}$, a downsized matrix ($n \times q$ instead of $n \times m$) is obtained to describe the nodes of the community.

$$\hat{\mathbf{U}} = \{u_{i,j} | 1 \leq i \leq n,\ 1 \leq j \leq q\} \qquad (4)$$

By extending Eq. (3) to the rank-q of $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$, the equation is written as

$$\hat{\mathbf{A}} = \hat{\mathbf{U}} \times \hat{\mathbf{\Sigma}} \times \hat{\mathbf{V}}^{\mathbf{T}} \qquad (5)$$

where $\hat{\mathbf{U}}$ is the $n \times q$ matrix of eigenvectors of $\mathbf{A}.\mathbf{A^T}$ and $\hat{\mathbf{V}}^{\mathbf{T}}$ is the $q \times n$ matrix of eigenvectors of $\mathbf{A^T}.\mathbf{A}$. The matrix $\hat{\mathbf{A}}$ is the approximate version of the node-term matrix. It should be noted that the columns of the new matrix $\hat{\mathbf{U}}$ have no direct relation to the dictionary and does not have any verbal meaning. They are actually merged version of most important terms regarding to each node.

The node-term matrix $\mathbf{A}$ represents the association between people and terms. By applying LSI to $\mathbf{A}$, we consider truncated node-term matrix $\hat{\mathbf{A}}$ as new model to describe the nodes by newly extracted features. One easy approach to extract the relationship between the nodes is calculating pair-wise similarities. If $a(i)$ and $a(j)$ are two feature vectors

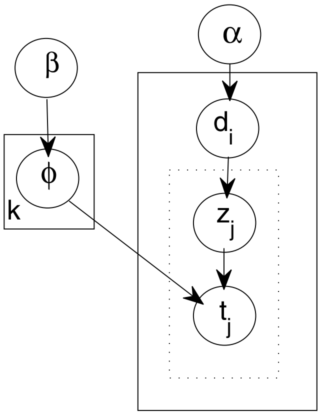


Figure 1: The graphical model of LDA.

(i^{th} and j^{th} rows of $\hat{\mathbf{A}}$) assigned to the i^{th} and j^{th} nodes of the community, the most popular similarity measure is the Cosine distance and is calculated as follows

$$\cos(i,j) = \frac{a(i).a(j)}{\|a(i)\|.\|a(j)\|} = \frac{\sum_{k=1}^{q} a(i,k).a(j,k)}{\sqrt{\sum_{k=1}^{q} a^2(i,k).\sum_{k=1}^{q} a^2(j,k)}} \tag{6}$$

The result is an $n \times n$ symmetric similarity matrix $\mathbf{S}$ in which every entry shows the similarity degree between two nodes that it can be interpreted as degree of relation between the nodes.

$$\mathbf{S} = \begin{pmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,m} \\ s_{2,1} & s_{2,2} & \cdots & s_{2,m} \\ \vdots & & & \\ s_{m,1} & s_{m,2} & \cdots & s_{m,m} \end{pmatrix}$$

Since a link between two nodes is usually represented by a binary value (linked = 1 and not link = 0), $\mathbf{S}$ has to be converted to a binary matrix. The process of converting non-binary similarity matrix into binary adjacency matrix is discussed in Section 5.

4.1 Link Prediction Using Hidden Topics

Latent Dirichlet Allocation (LDA) is a topic model and was originally presented as a graphical model for topic discovery and document representation [1]. LDA is a generative probabilistic model that describes sets of observations by unobserved data. LDA suggests that every document is a mixture of hidden topics. Each topic itself is a distribution over a set of words or dictionary. In other words, it assumes that every word in a document is generated by its own topic and relates all document to words through those latent topics. Similar to Probabilistic Latent Semantic Analysis (PLSA), in LDA, each document is viewed as a mixture of topics. However, in LDA the topics are assumed to have a Dirichlet distribution.

In document modeling using LDA, there are three essential elements including documents, terms, and topics. Every document $\mathbf{d} = \{d_1, d_2, ..., d_n\}$ is a sequence of terms from the dictionary $\mathbf{t} = \{t_1, t_2, ..., t_m\}$. A topic ϕ_i from $\phi = \{\phi_1, \phi_2, ..., \phi_k\}$ is a probability distribution over the dictionary $\mathbf{t}$ and models particular groups of terms that occur frequently in similar documents. Figure 1 depicts the graphical model of LDA. Each document d_i is indirectly related to the dictionary $\mathbf{t}$ through hidden variable $\mathbf{z}$ which is called topic assignment of words. α and β are called Dirichlet prior on the document and topic distribution. The hidden structure of topics is described by the posterior distribution of the hidden variable $\mathbf{z}$:

$$P(\mathbf{d}, \mathbf{z}, \phi | \mathbf{t}, \alpha, \beta) = \frac{P(\mathbf{t}, \mathbf{d}, \mathbf{z}, \phi | \alpha, \beta)}{\sum_{\phi_{1:k}} \sum_{d_{1:n}} P(\mathbf{t} | \alpha, \beta)} \tag{7}$$

In LDA model, the goal is to estimate the following probabilities: (i) the probability of topics given terms $P(\phi|\mathbf{t})$; (ii) the probability of documents given terms $P(\mathbf{d}|\mathbf{t})$; and (iii) the probability of topic assignments of terms $P(\mathbf{z}|\mathbf{t})$. Since estimating the posterior distribution on (7) is intractable, Gibbs sampling which simulates a high dimensional probability distribution by iteratively sampling one dimension at a time, is employed [5].

The Gibbs sampler generates two distributions from the node-term matrix $\mathbf{A}$. The two new structures are $\mathbf{D}$ and $\mathbf{W}$ which are called node-topic and word-topic matrices, respectively. Each entry $D(i,j)$ represents the number of times a term used by node i has been assigned to the topic j. An entry $W(k,j)$ is the number of term k has been assigned to the topic j. As a result the textual representation of the nodes is approximately modeled by the following equation:

$$\hat{\mathbf{A}} = \mathbf{D} \times \mathbf{W}^{\mathbf{T}} \tag{8}$$

where $\mathbf{D}$ is matrix representation of a bipartite graph linking nodes to topics. Because topics are co-occurred when representing the nodes, we can assume there is a semi-bipartite graph represented by $\mathbf{Y}$:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{R} & \mathbf{D} \\ \mathbf{D}^{\mathbf{T}} & \mathbf{D}^{\mathbf{T}} \times \mathbf{D} \end{bmatrix} \tag{9}$$

where $\mathbf{Y}$ is a $(n+K) \times (n+K)$ structure consists of four blocks. Here $K = 100$ is the number of topics.

In a bipartite graph, nodes are divided into two disjoint groups. While there are edges from one group of nodes to the other and vise versa, there is no links between nodes inside a group. Unlike bipartite graphs, in a semi-bipartite graph, nodes in one of groups can be linked to each other. The structure that has been introduced in (9) is a semi-bipartite graph. Figure 2 depicts a real semi-bipartite graph extracted from our data set. While the edges between authors are missing, the edges between two groups (authors to topics and vise versa) have been already learned by LDA modeling. In addition, the links between topics ($\mathbf{D}^{\mathbf{T}} \times \mathbf{D}$) which are simply topic co-occurrence relations can be calculated.

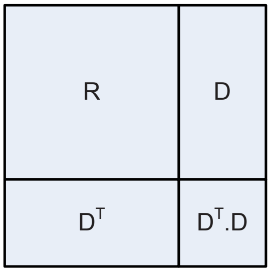


Figure 2: Author-topic semi-bipartite adjacency matrix.

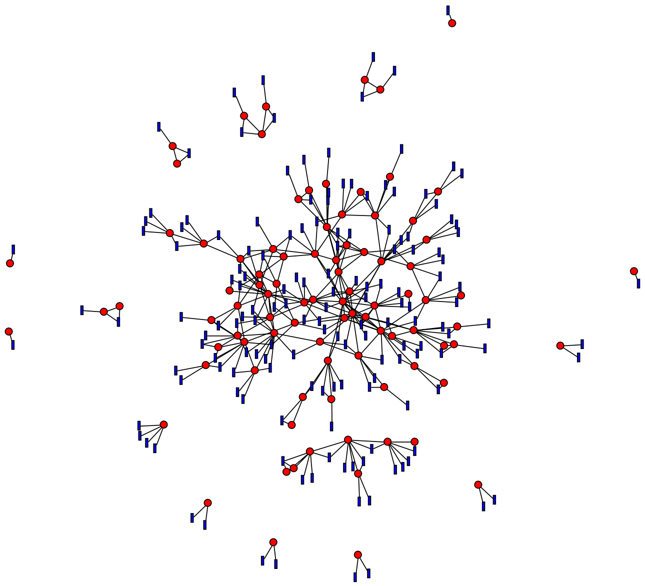


Figure 3: An example of an author-topic semi-bipartite network. Square nodes depict authors and round nodes represent topics.

Out of four blocks of the matrix $\mathbf{Y}$, three blocks are already calculated. The only unknown block is $\mathbf{R}$ which represents the author-author relations. In other words, $\mathbf{Y}$ is partially revealed structure. Using topological features such as short path, common neighbors and so on, we can predict the missing entries in $\mathbf{Y}$ which are essentially co-authorship relations (Figure 2).

The core idea in the proposed approach is instead of using similarity approach based on node attribute features (BOW representation of the authors) we use topological measures to predict social links. One problem with the similarity approach is that we have to convert a non-binary similarity matrix to a binary adjacency matrix. A trivial approach is to apply a threshold to the non-binary similarity matrix. According to the results reported in this paper, this approach offers very poor precision which is more important than recall in most social networks problems.

Among many topological metrics measures [10] which use topological features of the graph, Katz score has demonstrated better results [16]. The Katz score has been used in many ranking, link prediction, and clustering problems. This score which is calculated for any pair of nodes, is estimated by summation of collection of paths between two nodes:

$$Katz(y_i, y_j) = \sum_{L=1}^{\infty} \alpha^L |path_L(y_i, y_j)| \tag{10}$$

where y_i and y_j are two nodes of the semi-bipartite graph, $\alpha \leq 1/||Y||_2$ is damping factor, and $path_L(y_i, y_j)$ is the set of all length$-$L paths between y_i and y_j. To calculate Katz score for whole matrix $\mathbf{Y}$ we have:

$$Katz(\mathbf{Y}) = \alpha\mathbf{Y} + \alpha^{\mathbf{2}}\mathbf{Y}^{\mathbf{2}} + ... = (\mathbf{I} - \alpha\mathbf{Y})^{\mathbf{-1}} - \mathbf{I} \tag{11}$$

Katz score is calculated for all pairs of block $\mathbf{R}$ in matrix $\mathbf{Y}$ which address co-authorship links.

5. EXPERIMENTAL RESULTS

Co-authorship data of 20 scientific domains as described in section 2 are collected. The proposed method which is based on predicting links using topological features of a semi-bipartite graph inferred from the training documents is compared with similarity-based predictions. In the proposed methods, along with Katz, Short Path (SP) score is also implemented. For similarity-based method which works with feature vectors, two feature vector representations are used: BOW and LSI-based representation. To calculate similarities, Cosine distance is calculated for each pair of nodes. In Tables 2 and 3, the results of two methods are illustrated.

All methods are evaluated in two different scenarios: (i) recommending single link for each node, (ii) extracting the structure of social network (all links at the same time). For the first scenario, two IR evaluation methods Mean Reciprocal Rank (MRR) and Precession at first retrieved link (P@1) are estimated. As Table 2 illustrates, BOW representation and similarity method works very well for recommending single link to every node and we do not need to implement complex techniques such as LSI and LDA. However, if the task is extracting the structure (all links) of a social network, which is more realistic scenario, neither BOW nor LSI representations offer high precision and recall at the same time. The problem is more crucial if we note that in most social network extraction tasks, high precision is required rather than high recall. Table 3 illustrates precision, recall, and F-measure for all methods. The proposed method offers high precision and consequently high F-measure compared to both similarity-based methods.

In the first scenario which is recommending new links to the nodes, the recommender system generates a set of candidates sorted by their link scores. In the second scenario, the goal is to generate the social structure and it requires determining the existence or absence of every single link. In other words, we have to infer a binary value from non-binary link score (estimated either similarity-based or Katz score) which addresses a thresholding problem. In this paper, the threshold value is automatically determined using the link probability distribution. We know that most social networks (including the co-authorship networks) are very sparse structures. It means that the probability of a link between two nodes is very small (0.01 to 0.10). In this paper, we assume that the sparsity of the networks is 0.95 which means the probability of a link is 0.05. Given this assumption, the threshold value $\hat{\tau}$ is estimated as follows:

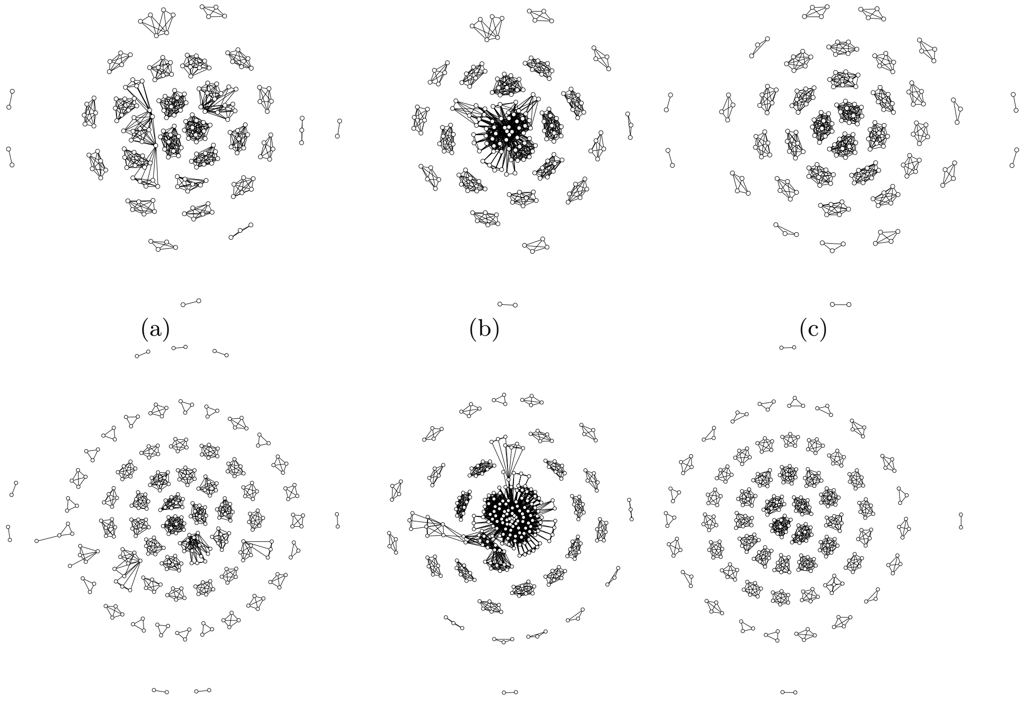
$$\hat{\tau} = \arg\min \left[\frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n} Th(n_i, n_j) - 0.05 \right] \tag{12}$$

$$Th(n_i, n_j) = \begin{cases} 0 : s(n_i, n_j) \leq \tau \\ 1 : s(n_i, n_j) > \tau \end{cases} \tag{13}$$

Figure 4 illustrates two co-authorship networks extracted using the similarity-based and topics-based models.

6. CONCLUSION

Many topological metrics for link prediction have been already proposed. Although some of these metrics such as Katz and rooted PageRank are very effective, they cannot be used if we want to build a network from scratch. It means to be able to predict social links based on topological metrics



(a) (b) (c)

(d) (e) (f)

Figure 4: Real and predicted "Neuroimaging" (a to c) and "Pathology" (d to f) co-authorship networks: (a and d) Original network; (b and e) Similarity-based; and (c and f) LDA-based prediction.

Table 2: P@1 and MRR of the network extraction methods on 20 co-authorship network data.

	MRR				P@1			
id	BOW	LSI	LDA+K	LDA+SP	BOW	LSI	LDA+K	LDA+SP
1	1.000	1.000	0.763	0.763	0.921	0.887	0.752	0.752
2	1.000	1.000	0.818	0.818	0.929	0.835	0.851	0.850
3	1.000	1.000	0.693	0.690	0.951	0.779	0.739	0.738
4	1.000	1.000	0.859	0.859	0.934	0.847	0.839	0.838
5	1.000	1.000	0.950	0.950	0.838	0.735	0.832	0.831
6	1.000	1.000	0.889	0.889	0.823	0.709	0.774	0.773
7	1.000	1.000	0.763	0.763	0.921	0.887	0.752	0.752
8	1.000	1.000	0.872	0.872	0.875	0.720	0.867	0.866
9	0.996	0.996	0.604	0.572	0.914	0.798	0.606	0.588
10	0.995	0.995	0.813	0.813	0.888	0.769	0.781	0.779
11	1.000	1.000	0.789	0.789	0.957	0.815	0.767	0.764
12	1.000	1.000	0.960	0.960	0.891	0.733	0.883	0.883
13	1.000	1.000	0.889	0.889	0.823	0.709	0.774	0.773
14	0.997	0.997	0.657	0.657	0.897	0.857	0.678	0.678
15	1.000	1.000	0.874	0.874	0.954	0.815	0.881	0.880
16	1.000	1.000	0.902	0.902	0.949	0.723	0.792	0.791
17	0.996	0.996	0.625	0.625	0.837	0.786	0.546	0.545
18	0.996	0.996	0.506	0.502	0.863	0.841	0.480	0.475
19	1.000	1.000	0.888	0.888	0.910	0.743	0.889	0.889
20	1.000	1.000	0.791	0.791	0.891	0.784	0.827	0.826
Average	**0.999**	**0.999**	0.795	0.793	**0.898**	0.789	0.765	0.764

we need some existing links or partially revealed network. In this paper, a new approach to predicting social links based on extracting hidden topics from text data is proposed. To utilize topological metrics, first the nodes are associated with hidden topics extracted by LDA, and second, the generated semi-bipartite structure is convected into a regular graph in which node-node relations are missing. By employing topological metrics such as Katz and short path, missing links are predicted with high precision.

7. REFERENCES

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

[2] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[3] J. Diesner and K. M. Carley. Exploration of

Table 3: Precision, Recall, and F-measure of the network extraction methods on 20 co-authorship network data.

	Precision				Recall				F-measure			
id	BOW	LSI	LDA+K	LDA+SP	BOW	LSI	LDA+K	LDA+SP	BOW	LSI	LDA+K	LDA+SP
1	0.184	0.213	0.925	0.624	0.997	0.997	0.799	0.799	0.311	0.350	0.858	0.701
2	0.316	0.273	0.907	0.767	1.000	1.000	0.955	0.955	0.481	0.428	0.931	0.851
3	0.162	0.188	0.795	0.625	0.995	0.992	0.879	0.879	0.279	0.316	0.835	0.730
4	0.294	0.326	0.925	0.734	0.997	0.997	0.845	0.845	0.453	0.491	0.883	0.786
5	0.388	0.447	0.962	0.808	1.000	1.000	0.861	0.861	0.559	0.618	0.909	0.834
6	0.308	0.334	0.950	0.752	0.987	0.987	0.758	0.758	0.469	0.499	0.843	0.755
7	0.184	0.213	0.925	0.624	0.997	0.997	0.799	0.799	0.311	0.350	0.858	0.701
8	0.264	0.268	0.869	0.751	0.981	0.981	0.887	0.887	0.416	0.421	0.878	0.813
9	0.196	0.225	0.773	0.534	0.935	0.935	0.548	0.548	0.324	0.363	0.641	0.541
10	0.403	0.413	0.959	0.740	0.911	0.911	0.684	0.684	0.559	0.568	0.798	0.711
11	0.795	0.855	0.995	0.873	1.000	1.000	0.823	0.823	0.886	0.922	0.901	0.847
12	0.655	0.662	0.985	0.832	0.980	0.980	0.855	0.855	0.785	0.790	0.915	0.843
13	0.308	0.334	0.950	0.752	0.987	0.987	0.758	0.758	0.469	0.499	0.843	0.755
14	0.126	0.142	0.695	0.500	0.965	0.965	0.788	0.788	0.223	0.248	0.739	0.612
15	0.322	0.321	0.893	0.738	1.000	1.000	0.930	0.930	0.487	0.486	0.911	0.823
16	0.514	0.621	0.993	0.824	1.000	1.000	0.650	0.650	0.679	0.766	0.786	0.727
17	0.191	0.199	0.876	0.574	0.943	0.937	0.648	0.648	0.317	0.328	0.745	0.609
18	0.100	0.105	0.844	0.405	0.927	0.927	0.463	0.463	0.181	0.190	0.598	0.432
19	0.262	0.299	0.861	0.738	1.000	1.000	0.939	0.939	0.415	0.460	0.898	0.826
20	0.281	0.280	0.819	0.673	1.000	1.000	0.898	0.898	0.439	0.438	0.857	0.769
Average	0.313	0.336	**0.895**	0.693	**0.980**	0.980	0.788	0.788	0.452	0.477	**0.831**	0.733

communication networks from the enron email corpus. In *Proc. of Workshop on Link Analysis, Counterterrorism and Security at SIAM International Conference on Data Mining 2005. Newport Beach, CA, April 21-23, 2005*, pages 3–14, 2005.

[4] D.Watts and S. Strogatz. Collective dynamics of smallworld networks. *Nature*, (363):202–Ű204, 1998.

[5] 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, April 2004.

[6] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning, 2006.

[7] A. E. Hassan and R. C. Holt. The small world of software reverse engineering. In *WCRE '04: Proceedings of the 11th Working Conference on Reverse Engineering (WCRE'04)*, pages 278–283, Washington, DC, USA, 2004. IEEE Computer Society.

[8] R. Homayouni, K. Heinrich, L. Wei, and M. W. Berry. Gene clustering by latent semantic indexing of medline abstracts. *Bioinformatics*, 21(1):104–115, 2005.

[9] D. Jensen and J. Neville. Data mining in social networks. *National Academy of Sciences Symposium on Dynamic Social Network Analysis, November 7-9, 2002, Washington, DC: National Academy Press.*, 2002.

[10] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proceedings of the twelfth international conference on Information and knowledge management*, CIKM '03, pages 556–559. ACM, 2003.

[11] N. Matsumura, D. Goldberg, and X. Llora. Mining directed social network from message board. In *In WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web, New York, NY, USA, 2005. ACM Press.*, pages 1092–1093, 2005.

[12] Y. Matsuo, H. Tomobe, K. Hasida, and M. Ishizuka. Mining social network of conference participants from the web. In *WI '03: Proceedings of the IEEE/WIC International Conference on Web Intelligence*, pages 190–193, Washington, DC, USA, 2003. IEEE Computer Society.

[13] J. Mori, Y. Matsuo, M. Ishizuka, and B. Faltings. Keyword extraction from the web for foaf metadata. *1st Workshop on Friend of a Friend, Social Networking and the Semantic Web, 1-2 September 2004, Galway, Ireland*, 2001.

[14] J. Resig and A. Teredesai. A framework for mining instant messaging services. In *In Proceedings of the 2004 SIAM DM Conference*, 2004.

[15] M. Richardson and P. Domingos. Mining knowledge-sharing sites for viral marketing. In *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 61–70, New York, NY, USA, 2002. ACM Press.

[16] Z. Yin, M. Gupta, T. Weninger, and J. Han. Linkrec: a unified framework for link recommendation with user attributes and graph structure. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 1211–1212, New York, NY, USA, 2010. ACM.