Supervised Prediction of Social Network Links Using Implicit Sources of Information

Ervin Tasnádi
Institute of Informatics
University of Szeged
Szeged, Hungary
Tasnadi.Ervin@stud.u-szeged.hu

Gábor Berend
Institute of Informatics
University of Szeged
Szeged, Hungary
berendg@inf.u-szeged.hu

ABSTRACT

In this paper, we introduce a supervised machine learning framework for the link prediction problem. The social network we conducted our empirical evaluation on originates from the restaurant review portal, *yelp.com*. The proposed framework not only uses the structure of the social network to predict non-existing edges in it, but also makes use of further graphs that were constructed based on implicit information provided in the dataset. The implicit information we relied on includes the language use of the members of the social network and their ratings with respect the businesses they reviewed. Here, we also investigate the possibility of building supervised learning models to predict social links without relying on features derived from the structure of the social network itself, but based on such implicit information alone. Our empirical results not only revealed that the features derived from different sources of implicit information can be useful on their own, but also that incorporating them in a unified framework has the potential to improve classification results, as the different sources of implicit information can provide independent and useful views about the connectedness of users.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Data Mining

Keywords

Link prediction; Social networks; random walk with restarts; Yelp

1. INTRODUCTION

User-generated contents, including social media, is among the primary sources of information nowadays. For instance, customers tend to obtain other people's opinion about certain products and services via review sites, such as yelp.com. Review portals and other social media platforms often allow their users to follow or add other users to their followee or contact lists, naturally forming social networks in this way. Edges connecting two users in these networks often reflect the mutual interest of the users. In the case of thematic portals, including restaurant review sites, the underlying social structure can help users to find other users of similar interest and taste, whose opinions they might want to pay attention to in the future. As there are plentiful of users one can follow in social networks, it is thus desirable for users to get automatic suggestions on who they might be interested to follow or add as a friend. As it is also possible, that some portals do not explicitly allow their users to form a social network, we also investigate such models in this work, which do not rely on the connections between the users at all upon trying to predict whether there exists a link in the original social network between two users. These models rely on the similarity in the user-item ratings and the language use of the users for prediction.

Link prediction is a common research task related to social networks, which formalizes the question "can we infer which new interactions among its members are likely to occur in the near future?" [7]. In these works the social networks are defined as the collection of nodes representing (potentially) different types of entities (e.g. customers and businesses) and the edges express some kind of relation – such as influence, collaboration or purchase – between them. These graphs often take special bipartite or n-partite forms, however, some work use the projected version of such graphs [1].

In our work, similarly to the method proposed in [3], we formally treated link prediction as a classification task, where the task is to decide whether the friendship relation holds between a pair of users. This paper makes the following contributions to the topic of link prediction:

- Upon predicting links in a social network graph, we investigated the applicability of features that do not rely directly on the social network. Instead, we derived features from bipartite graphs containing implicit information about the members of the social network.
- We empirically evaluated and compared the performance of models that purely rely on implicit information about the users and models that had access to the social network itself.
- We propose new random walks-based features, by defining the 'distance' between pairs of users in various ways, e.g. as the Kullback-Leibler divergence of the

Copyright is held by the International World Wide Web Conference Committee (IW3C2). IW3C2 reserves the right to provide a hyperlink to the author's site if the Material is used in electronic media.
WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2743037 .

stationary distributions of the random walks that are rooted[1] in one of the users.

2. RELATED WORK

The prediction of links from (social) networks has an extensive literature, thanks to its wide-range applicability including the detection of possible terrorist cells [6] and the prediction of author collaborations [7]. This section briefly introduces some of the existing works that aim at solving the problem of link prediction.

The seminal work of [7] deals with the problem of forecasting future collaborations between scientists based on the snapshot of their collaboration network. This type of link prediction is sometimes called temporal link prediction as it takes into consideration the temporally evolving nature of social networks. There are articles, however, in which the identification of missing links is performed irrespective of temporal aspects, for which reason these are often referred to as structural link prediction tasks [11]. Our work belongs to the latter class of link prediction problems, as timestamps of the formation of friendships were not included in the dataset we experimented with.

A frequent way to solve link prediction is to calculate various topological metrics of the nodes within the network, which then serve as a basis for the calculation of the similarity of the pairs of nodes. The underlying assumption of such approaches is that the more similar two nodes are, the more likely they become connected. Common metrics derived from (social) networks include the number of common neighbors, or the length of the shortest path between pairs of nodes [7].

Besides ranking approaches, link prediction can also be modeled as a supervised learning task. A typical approach when applying supervised learning for link prediction is to calculate the kind of similarity scores that ranking approaches rely on and feed them to a classification algorithm [1]. The benefit of such approaches is that they can easily incorporate multiple discriminating factors at a time. Furthermore, defining features which describe the pair of nodes from a perspective other than the social network itself is straightforward.

It has been shown that substantial improvement can be achieved in link prediction by designing features based on the meta-data available about the nodes [3]. More specifically, as illustrated in [3], the sum of the articles written by a pair of authors can be fruitfully utilized to improve the performance of link prediction in a co-authorship network. One of their assumptions was that authors are more likely to collaborate in the future if they had written many articles previously (independent of each other). The work of [15] also argues that co-authorship prediction can be improved by relying on heterogeneous networks, i.e. networks capable of modeling relations between different types of entities, e.g. authors and conference venues. The shared meta-data of social network users was studied in [14], where it was empirically shown that users who generated content with similar tags were more likely to become friends on *flickr* and *last.fm*.

Our framework relates to these works as we also rely on features other than the ones that can directly be extracted from the social network itself. The difference of our approach to previous works relies in that the implicit features – that are not directly derived from the social network – are calculated based on bipartite graphs that are likely to be influenced by the social network.

Link prediction can also be viewed as a task suitable for recommendation systems. From this point of view, the task can be formulated to recommend users such ‘items’ which are themselves further users as well. Matrix factorization techniques are particularly popular in the field of recommendation systems [2, 5, 12].

3. THE PROPOSED FRAMEWORK

As mentioned earlier, we treated the classification of potential edges in the social network of the Yelp Challenge dataset as a supervised binary learning task. In our framework, the feature space comprises of features deriving from different feature groups, which serve as different ‘views’ of the classification instances. During the design of the features, our intention was to describe the similarity of the users from different aspects, namely

- the similarity of the language use of their reviews,
- the restaurants they visited and
- their proximity in the social network.

The graphs corresponding to the above three aspects are to be introduced subsequently.

3.1 Auxiliary graphs

According to the different aspects, we constructed auxiliary graphs from which we derived the features for our supervised classification framework. This section introduce these auxiliary graphs.

3.1.1 User-Word graph

In the *User-Word (bipartite) graph* two user-type nodes were connected through a word-type node if the two users – corresponding to the user-type nodes – used the same word – corresponding to the word-type node – in any of their reviews.

User-Word graph. Let $G = (V_U \cup V_W, E)$ be an undirected bipartite graph, where $U = \{u_1, \dots, u_n\}$ is the set of users of the social network, $V_U = \{v_{u_1}, ..., v_{u_n}\}$ is the set of the user-type nodes, $W = \{w_1, \dots, w_m\}$ is the set of indicator words, and $V_W = \{v_{w_1}, ..., v_{w_m}\}$ is the set of the word-type nodes. For every user u_i, we assign node $v_{u_i} \in V_U$ and for every processed word $w_k \in W$, we introduce the node $v_{w_k} \in V_W$. The edge $(v_{u_i}, v_{w_k}) \in E$ exists if and only if user u_i used a word that got mapped to w_k (during the preprocessing phase of the reviews) at least once in at least one of its reviews.

The motivation behind analyzing the users based on their vocabulary was based on our assumption that users whose topics of interest overlaps substantially tend to use similar words in their reviews. If the words that are used by a pair of users do not overlap at all, it is reasonable to assume that they do not have much in common, hence, are less likely to become friends. On the contrary, if two users share multiple words, e.g. *pasta*, *pizza*, *pepperoni*, they seem to have a common passion towards Italian cuisine, making them more likely to be involved in a friendship relation. Similarly, if two users describe restaurants from similar aspects, using similar

[1]the term rooted random walk is sometimes also referred as random walk with restarts (RWR)

vocabulary, including e.g. words about the *politeness of the service* or the *ambiance of the restaurant*, this can indicate that the two users regard similar things as important, hence they might have a higher chance of becoming friends.

The construction of the User-Word graph had the following main steps. In order to eliminate word-type nodes of marginal relevance, we performed stop word filtering of the reviews and also discarded all words that were not tagged as nouns by the Stanford CoreNLP pipeline [9]. Keeping only words that were tagged as nouns seemed to provide a compromise between the number and the usefulness of the word-type nodes included in this graph. We thought nouns to be useful as most food names comprise of words that should be tagged as nouns. To further decrease the number of word-type nodes, word forms were also Porter-stemmed, so that some of the different word forms were then possible to be treated identically.

3.1.2 User-Restaurant graphs

A further aspect we took into consideration for modeling the users was based on the restaurants they visited. We defined three versions of the *User-Restaurant (bipartite) graphs* made up of *user*,- and *restaurant-type* nodes. One of the graphs expressed the *visited* relation between users and restaurants. In this graph, there existed a path of length two between a pair of users if there was at least one restaurant that was visited by both of them.

User-Restaurant graph. Let $G = (V_U \cup V_R, E)$ be an undirected bipartite graph, where V_U is the same as in the User-Word graph, $R = \{r_1, \ldots, r_l\}$ is the set of restaurants and V_R is the set of the restaurant-type nodes. For every user $u_i \in U$, we assign node $v_{u_i} \in V_U$ and for every restaurant $r_j \in R$, we introduce node $v_{r_j} \in V_R$. The edge $(v_{u_i}, v_{r_j}) \in E$ exists if and only if the user u_i visited the restaurant r_j at least once.

The reason for modeling users based on the restaurants they wrote a review about is based on the natural assumption that if two users tend to visit the same restaurants then their preferences are likely to be similar, making it more probable that they form a friendship. In order to confirm this assumption, we calculated the average number of restaurants for which both members of the user pairs wrote a review about. We calculated this amount for the user pairs who were friends of each other and for those user pairs for which the relation did not hold (according to the dataset), and got the results of 0.679 and 0.003, respectively.

We constructed two further bipartite graphs involving restaurants. While the previous graph contained the fact if a user visited and wrote a review about a restaurant, the aim of these graphs was to capture the users' satisfaction towards the restaurants. We measured the users' (dis)satisfaction by taking into consideration their star ratings; a simple baseline predictor – commonly applied in the field of collaborative filtering [4] – was used during the construction of these graphs.

In one of the graphs two users were connected through a restaurant only if they had a common positive opinion about it, while for the other graph, two users were connected through a restaurant node only if they had a common negative feeling towards it. The (dis)satisfaction of user u_i towards restaurant r_j was determined by comparing the actual rating $r(u_i, r_j)$ to the predicted rating of the baseline predictor, i.e. we regarded user u_i to be satisfied with restaurant r_j if the inequality

$$r(u_i, r_j) > avg + \Delta u_i + \Delta r_j \tag{1}$$

held, where avg is the average of all the ratings in the database, Δu_i is the difference between the average of the ratings given by user u_i and avg, and Δr_j is the difference between the average of the ratings given for restaurant r_j and avg. In case inequality (1) did not hold, we regarded u_i to be dissatisfied with r_j and the edge connecting node v_{u_i} with node v_{r_j} was only included in the bipartite graph modeling the *dissatisfied_with* relation in that case.

3.1.3 User-User graph

The third aspect of our investigation was based on the *User-User* or *Social Network graph*. In this graph, nodes represented users and an edge connecting two nodes indicated that the users corresponding to the nodes were known to be friends of each other according to the dataset.

Social Network graph. Let $G = (V_U, E)$ be an undirected graph where V_U is the same as in the previous graphs. The edge $(v_{u_i}, v_{u_j}) \in E \subset V_U \times V_U$ exists if and only if the $friends(u_i, u_j)$ relation holds for users u_i and u_j.

This third – and most trivial – way to analyze users thus took place via the inspection of the social network itself. Relying on the social network to predict missing links from it is the most straightforward way to go, which has extensively used in previous works. We also used the information residing in the social network, however, it is important to note that one of our main research goals was to investigate and compare the performance of such frameworks which do not rely on social network information. We believe it is an important task, as there might be situations when it would be desirable to predict social links in such cases when even a partially observable version of the social network is difficult or even impossible to obtain.

3.2 Features

As illustrated above, it is common in all of our feature aspects that they can naturally be represented as graphs, thus the way features were extracted from them could be treated in a unified manner. In this section, we introduce various ways how features were derived from the graphs introduced in Section 3.1.

First, for every node of our interest, we calculated the stationary distribution of its rooted random walk. Rooted random walks – also referred to as random walks with restarts (or RWR for short) – simulate a random walk similar to PageRank [13]. Both RWR and PageRank algorithms contain a parameter β making the random surfer in any time to choose a node to traverse to from its direct neighbors with probability β. The difference of the RWR and PageRank algorithms lies in the determination of the subsequent node during the random walk with probability $1 - \beta$. More precisely, RWR returns to the dedicated node, i.e. the root node, while PageRank chooses any of the nodes of the graph uniformly at random with probability $1-\beta$. The above characteristic of RWR makes its stationary distribution available to be interpreted as a measure of similarity between the root node and the rest of the nodes. During our experiments, we applied the commonly used value of 0.8 for the parameter β.

Computing the similarity scores for a graph is an expensive task with the straightforward implementations, however, fast approximate approaches exist to calculate the stationary distribution of RWRs. The authors of [16], for instance, claim that their approach might benefit a 150-times speedup, while the approximate stationary distribution returned by their method preserves 90% of the quality of the optimal one.

Similarity score. For a graph $G = (V, E)$ and nodes $v_i, v_j \in V$, we say that $s_{i \to j} \in [0, 1]$, i.e. the similarity score of v_j to v_i, equals the stationary distribution of the RWR rooted in node v_i with respect node v_j.
We also define the *similarity vector* on the graph G for the root node v_i as $\mathbf{sim}_i = [s_{i\to 1}, ..., s_{i\to n}]$, that is simply the stationary distribution of the RWR rooted in node v_i.

Note that the way similarities are defined makes it possible to use any of the graphs introduced in Section 3.1 as G. This way we can generate a feature value to any pair of users (u_i, u_j) based on any of the graphs providing different views about them. Following, we specify the details how feature values were derived for the pairs of users. Due to the fact that our purpose was to define similarity between pairs of users, we pruned and renormalized the stationary distributions of RWRs calculated for bipartite graphs to include user-type nodes alone.

To measure the global similarity of the stationary distributions of the random walks with roots v_{u_i} and v_{u_j} regarding user pair (u_i, u_j), we computed the *Kullback-Leibler (KL) divergence* of the similarity vectors $\mathbf{sim}_i$ and $\mathbf{sim}_j$. Here, we expected that if two users behave similarly in the graph, then their similarity vectors tend to be similar, which results in their KL-divergence to tend to 0. As the Kullback-Leibler divergence is asymmetric, we derived features for both the value $D_{KL}(\mathbf{sim}_i \parallel \mathbf{sim}_j)$ and $D_{KL}(\mathbf{sim}_j \parallel \mathbf{sim}_i)$.

Besides the previous features – taking into account the entire stationary distributions of the random walks with restarts – we defined features that specifically considered the stationary distributions regarding nodes v_{u_i} and v_{u_j} for a given user pair (u_i, u_j). As one further type of feature, we introduced the rank-based similarity, which did not take into consideration the exact values of the similarity vectors, rather focused on the ranks of the nodes according to the stationary distribution of the RWRs.

Similarity rank. The similarity rank of node v_j in the similarity vector $\mathbf{sim}_i$ is defined as the number of the elements with a higher value of stationary distribution being assigned to them. The similarity rank of v_j in $\mathbf{sim}_i$ is thus $rank_{\mathbf{sim}_i}(v_j) = |\{k : s_{i\to k} > s_{i\to j}\}|$.

Based on the definition of rank similarity, we were then able to measure how 'close' two nodes v_i and v_j were from each other. For measuring rank similarity, we employed the formula

$$1 - \frac{rank_{\mathbf{sim}_{v_i}}(v_i) - rank_{\mathbf{sim}_{v_i}}(v_j)}{n},$$

where n is the number of users in the social network. Upon determining the feature values for a classification instance describing a user pair (u_i, u_j), we also calculated the above formula as

$$1 - \frac{rank_{\mathbf{sim}_{v_j}}(v_j) - rank_{\mathbf{sim}_{v_j}}(v_i)}{n},$$

in order to account for the similarity of the friendship relation. A value for that feature being close to 1 is intended to mean that the two users for which the feature was calculated are similar.

4. EXPERIMENTAL RESULTS

In this section, we provide statistics about the dataset and the graphs we derived from it, and also display our empirical evaluation scores.

4.1 The dataset

The dataset we evaluated our approaches on is the Yelp Dataset Challenge[2] (4th edition) containing 42,153 restaurants, 252,898 users and 1,125,458 reviews. The dataset contains 955,999 pairs of users who are friends of each other. There are further information – such as check-ins and business attributes – in the dataset, that we did not utilize in our approach yet, however, we are planning to do so in the future.

As the dataset contained the reviews in a convenient format, it allowed us to construct the graphs as described in Section 3.1. The User-Word graph that was built from the preprocessed and filtered contents of the reviews, contained 97,705 word-type nodes besides the user-type ones, and it had more than 34.65 million edges. The User-Restaurant and the social network graphs had edge counts above 2.17 and 1.91 million, respectively. The two further (dis)satisfaction graphs both had approximately half the number of the edges of the original User-Restaurant graph they were derived from. For graphs involving user-, and restaurant-types entities, the number of nodes was directly influenced by the number of users and restaurants provided in the dataset.

4.2 Results

We now introduce our experimental settings in more details and also provide our baseline results. As stated previously, our task was to build a model which is able to decide whether two users should be connected in the social network. In order to build a training corpus, we randomly selected 1,000 pairs of users for which the friendship relation held and another 1,000 pairs of users, who were not friends of each other. Since there were no timestamps regarding the formulation of friendships, we did not guide the selection of the 1,000 edges that we deleted from the social network. Further 1,000 user pairs not being friends in the original social network were also selected randomly. These 1,000-1,000 samples then formed our list of instances belonging to the positive and negative class, respectively.

As links between users – who could otherwise be friends – might be absent as a result of the incompleteness of the social network, we can only be certain about the class labels of the positive instances. Due to the above observation and the fact that we are essentially interested in identifying instances belonging to the positive class, we only present the detailed performance measures (i.e. precision, recall and F-score) for the instances being labeled as positive. In order to get an overall measure of the classification performance, we present the accuracy of our classifiers as well.

4.2.1 *Baseline results*

Note that a random predictor would achieve an accuracy of 50% and an F-score of 0.5, as the dataset we used contained an equal number of user pairs belonging to the positive and negative instance classes. As such a baseline is rather simplistic, we also provide a matrix factorization-based approach as a baseline. For this, we relied on the

[2]accessible from http://www.yelp.com/dataset_challenge

d	10	20	50	100	125
Accuracy	0.545	0.540	0.653	0.661	0.662
Precision	0.545	0.540	0.616	0.625	0.625
Recall	0.542	0.537	0.811	0.801	0.807
F-score	0.543	0.538	0.700	0.702	0.705

Table 1: Baseline results as a function of the latent dimensions (d) used during the matrix factorization

Information	Accuracy	Precision	Recall	F-score
Word	0.737	0.732	0.750	0.739
Restaurant	0.780	0.755	0.830	0.790
Social	0.935	0.919	0.952	0.935

Table 2: Classification performance obtained relying on one source of information at a time

rating matrix $R \in [0 \dots 5]^{n \times m}$, the r_{ij} element of which is the star rating provided by user i with respect restaurant j. Our baseline applied the non-negative matrix factorization algorithm introduced in [8] to approximate the rating matrix as a product of matrices $W \in \mathbb{R}^{n \times d}$ and $H \in \mathbb{R}^{d \times m}$, so that $||R - WH||_F^2$ gets minimized and each element of W and H are non-negative. Choosing d in a way such that $d \ll m$ holds, the rows in $W \in \mathbb{R}^{n \times d}$ can be used as a lower-dimensional representation of the users in a 'latent' space. Due to its denser nature of the lower-dimensional representation, the comparison of the user pairs can be obtained more meaningfully as if it were performed in the original $m \gg d$ dimensional space (where most of the values in the vectors tend to be 0).

For this baseline, a user pair (u_i, u_j) was predicted as friends, if the following inequality held:

$$d_{cos}(\mathbf{w_{i:}}, \mathbf{w_{j:}}) \leq \frac{1}{n-2} \sum_{j' \notin \{i,j\}} d_{cos}(\mathbf{w_{i:}}, \mathbf{w_{j':}}), \qquad (2)$$

where n is the number of total users, $\mathbf{w_{i:}} \in \mathbb{R}^d$ denotes the ith row of W – that is the d-dimensional latent space representation of the user u_i – and the function $d_{cos}(\cdot, \cdot)$ refers to the cosine distance between two vectors. That is the user pair (u_i, u_j) was predicted to be friends, if their cosine distance did not exceed the average cosine distance of all the other users compared to user u_i. We tried to modify right side of inequality (2) in such a way that the average (or the maximum) of the cosine distances compared to user u_i are not taken with respect all the other users, but only for the friends of u_i, however, doing so did not result in better baseline performances.

As our baseline approach is sensible to the selection of d, the dimension of the row vectors in W, we experimented with various values of d. As seen from Table 1, the performance measures obtained by choosing low (i.e. ≤ 20) values of d are not much better than that of a random baseline. Although such small values of d perform poorly, performance measures seem to improve and stabilize once the reduced dimensionality of the user space is increased (i.e. $d \geq 50$). Table 1 also reveals that it is reasonable to think that increasing further the value of d above 100, do not yield substantial improvements in the results, as only a marginal improvement can be observed when increasing d from 100 to 125.

W	R	S	Accuracy	Precision	Recall	F-score
•	•		0.807	0.802	0.817	0.809
•		•	0.932	0.918	0.947	0.932
	•	•	0.934	0.924	0.945	0.934
•	•	•	0.932	0.921	0.945	0.933

Table 3: Classification results combining the different sources of information. Letters W, R and S refer to the word, restaurant and social graphs, respectively.

4.2.2 *Supervised learning-based results*

For the evaluation purposes, we built maximum entropy models relying on the feature space that were introduced in Section 3.2. Our models were trained using the Mallet machine learning framework [10]. In order to reduce the variability of our estimates for the performance of our models, we used 10 fold cross-validation during our experiments. As stated earlier, precision, recall and F-score metrics are presented for the positive class of instances – besides the overall classification accuracy.

Table 2 illustrates the classification performance obtained by relying on one source of information (i.e. word, restaurant or social) at a time. Due to the expectations, information about the social network turned out to be the most useful, while the model based on the language use of the users achieved the worst performance. The word usage features being the least informative, we should add that this approach still outperforms our matrix factorization based baselines, with a relatively large margin. Furthermore, we believe that there is still possibility to improve the word usage-based prediction of links, e.g. by relying on ontologies or incorporating word-type nodes in some more sophisticated manner. We plan to explore such extension possibilities in our future work.

We were also interested how the different sources of information about the users interact with each other. The result of these experiments, i.e. when more than one sources of information were used at a time, are included in Table 3. From this table, we can see that relying on more that just one source of information, we could improve our link prediction performances. This is especially true for relying on the words and restaurants-related informations at a time. Models which use the social graph as features did not really seem to differ from each other. This, however, could be anticipated, since the features derived from the social network itself are expected to serve extremely valuable information for link prediction. We should, however, emphasize that models not utilizing the structure of the social network at all, was able to achieve an F-score and accuracy above 0.8, that we regard as a promising result.

5. CONCLUSIONS

In this work, we proposed a supervised learning framework for the task of link prediction. The social network we evaluated our approach on was that of the restaurant review portal, yelp.com. In our proposed approach, we successfully exploited implicit sources of information, such as the language use of the reviewers and the restaurants they visited.

Thanks to the alternative sources of information, – not directly dependent on the structure of the social network itself – we managed to achieve reliable performances. In this paper, we also defined different ways to obtain similarity scores for pairs of users based on the stationary distribution of rooted random walks on different graphs. These similarity scores then proved to be useful in the supervised learning of link prediction.

In our future work, we are planning to explore further implicit sources of information about users to better approximate the link prediction performance that can be achieved by relying on the structure of the social network as well.

6. REFERENCES

[1] N. Benchettara, R. Kanawati, and C. Rouveirol. Supervised machine learning applied to link prediction in bipartite social networks. In *Proceedings of the 2010 International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '10, pages 326–330, Washington, DC, USA, 2010. IEEE Computer Society.

[2] K. Buza and I. Galambos. An application of link prediction in bipartite graphs: Personalized blog feedback prediction. In *8th Japanese-Hungarian Symposium on Discrete Mathematics and Its Applications June 4-7, 2013, Veszprém, Hungary*, page 89.

[3] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *In Proc. of SDM 06 workshop on Link Analysis, Counterterrorism and Security*, 2006.

[4] Y. Koren and R. Bell. Advances in collaborative filtering. In *Recommender systems handbook*, pages 145–186. Springer, 2011.

[5] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8):30–37, 2009.

[6] V. Krebs. Mapping networks of terrorist cells. *CONNECTIONS*, 24(3):43–52, 2002.

[7] 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, New York, NY, USA, 2003. ACM.

[8] C.-J. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural Comput.*, 19(10):2756–2779, Oct. 2007.

[9] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, 2014.

[10] A. K. McCallum. MALLET: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[11] A. K. Menon and C. Elkan. Link prediction via matrix factorization. In *Machine Learning and Knowledge Discovery in Databases*, pages 437–452. Springer, 2011.

[12] A. Mnih and R. Salakhutdinov. Probabilistic matrix factorization. In *Advances in neural information processing systems*, pages 1257–1264, 2007.

[13] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999.

[14] R. Schifanella, A. Barrat, C. Cattuto, B. Markines, and F. Menczer. Folks in folksonomies: Social link prediction from shared metadata. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM '10, pages 271–280, New York, NY, USA, 2010. ACM.

[15] Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han. Co-author relationship prediction in heterogeneous bibliographic networks. In *Proceedings of the 2011 International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '11, pages 121–128, Washington, DC, USA, 2011. IEEE Computer Society.

[16] H. Tong, C. Faloutsos, and J.-Y. Pan. Fast random walk with restart and its applications. In *Proceedings of the Sixth International Conference on Data Mining*, ICDM '06, pages 613–622, Washington, DC, USA, 2006. IEEE Computer Society.