1K1-IOS-1a-2

Exploiting Information Flow and Active Links for Link Prediction in Social Networks

Lankeshwara Munasinghe[*1] Ryutaro Ichise[*2]

[*1] Department of Informatics
The Graduate University for Advanced Studies
Tokyo, Japan
lankesh@nii.ac.jp

[*2] National Institute of Informatics
Tokyo, Japan
ichise@nii.ac.jp

Link prediction in social networks, such as friendship networks and coauthorship networks, has recently attracted a great deal of attention. There have been numerous attempts to address the problem of link prediction through diverse approaches. In the present paper, we focus on information flow in social networks and how it affects for the future link evolution. Particularly, the information flow not only depends on the link weights but also on the activeness of the links. The links become active if the interactions happen frequently and recently with respect to the current time. The time stamps of the interactions or links provide vital information for determining the activeness of the links. In the present paper, we introduced a new algorithm, referred to as *T_Flow*, that captures the important aspects of active links and information flow in social networks. We tested *T_Flow* with two social network data sets, namely, a data set extracted from Facebook friendship network and a coauthorship network data set extracted from *ePrint archives*. When we compare the link prediction performances with the previous method, the results of *T_Flow* method revealed that 0.17 improvement in F-measure for coauthorship data and 0.03 improvement in F-measure for Facebook data.

1. Introduction

Link prediction was introduced as a way to infer which new links are likely to occur in the near future in a given network [Liben-Nowell 03]. If we are presented with a snapshot of a network at the current time, the goal is to predict links that are to be occur in future. The structural information, features of nodes and edges of the given network can be used to predict future links.

Link prediction has many applications and it offers lot of benefits to the users of social networking services. For example, online social networking services, such as Facebook, can use link prediction to provide their users with better recommendations or suggestions. Therefore, users of these services can efficiently find their friends, colleagues, or people whom they wish to meet. Organizations such as research organizations, business organizations, and security agencies will be able to uncover information regarding unseen relationships among people or organizations. Thus, they may operate more effectively. Link prediction in scientific researcher networks allow researchers to find experts and research organizations in the same research field [Wohlfarth 08]. However, highly structured massive real-world networks involving heterogeneous entities with complex associations have added new challenges to link prediction research. Supervised and unsupervised learning methods [Kashima 09, Hasan 06] have been used in previous studies with different frameworks for link prediction. However the machine learning approaches remain an immense challenge due to different factors such as sparsity, complexity, size, time-dependent nature of the networks and imbalance between possible links and actual links observed in these networks [Lichtenwalter 10].

Information flow between nodes is a vital factor for link evolution in social networks. However, the information flow varies over time. It is worthy to study the factors which determine the information flow and how these factors can be effectively used for link prediction in networks. Particularly, the active links are one of the key factor which determines the information flow. The information flows from one node to the other through the set of links which connects them. This set of links is considered as the paths between the pair of nodes. Some of the recent link prediction researches have introduced unsupervised methods based on information flow in networks. One of the recent algorithm *PropFlow* [Lichtenwalter 10] has used random walk with link weights as the transition probabilities for the random walker. The supervised random walk algorithm introduced in [Backstrom 11] assign strengths to edges using edge and node attributes. The random walker use the strengths as the transition probabilities. However, those studies have not been considered the activeness of the links when random walker selects its path. We therefore introduced a new method which considers the effect of link weights as well as active links for information flow.

The remainder of the present paper is organized as follows. We discuss related research and the importance of activeness of links and information flow for link prediction under related research section. In the following section, we introduce a method of link prediction and the newly proposed algorithm, *T_Flow*. Experimental results are presented in experimental evaluation section followed by general discussion. Final section presents our conclusions.

2. Related work

In this section, we review research related to link prediction as well as background information on link prediction.

The increase in the number of studies related to link prediction in the recent past reveals the emerging interest in link prediction. Diverse approaches, including machine learning approaches and probabilistic approaches, have been proposed in order to address the problem of link prediction.

Supervised and unsupervised machine learning technologies have been used for link prediction. Classification using a learned model is the prominent feature of machine learning. A set of structural features of networks introduced in [Liben-Nowell 03] have been widely used with machine learning methods in the past researches such as [Pavlov 07] for link prediction in coauthorship networks. Later, the introduction of new features such as cooccurrence probability [Wang 07], keyword match count for paper topics and abstracts [Sachan 10] in combination with supervised machine learning methods provided more accurate link predictions in coauthorship networks. the number of links. These previous studies have proven the consistency and effectiveness of machine learning methods in link prediction.

Besides machine learning and probabilistic approaches, other different approaches can be seen in the literature. Parametric probabilistic model based on topological features of networks has introduced in [Kashima 06] for link prediction in biological networks. A new approach has introduced in [Sharan 08] to model both temporal and relational dependencies in the data. A supervised learning approach introduced for predicting link strengths using transactional information by [Kahanda 09] shows the correlation between varying link strength and future link evolution. A matrix alignment method was used to determine the most predictive features of a link structure by aligning adjacency matrix of a network with weighted similarity matrices [Scripps 08]. The weighted similarity matrices computed from node attributes and neighborhood topological features. The weights were learned by minimizing an objective function.

Some of the above worthy studies have considered how information flow through active links affects link evolution in the networks but most others are not. The recent research by [Munasinghe 12] has introduced a new feature which captures the impact of active links and active nodes. However, it is limited to common neighbors. The *PropFlow* [Lichtenwalter 10] algorithm allows to define the neighborhood of a node and not limited to common neighbors. However, it has not considered the activeness of the links when random walker selects its path. We therefore extended *PropFlow* algorithm in order to consider the effect of link weights as well as active links for information flow.

3. Supervised learning method for link prediction

Most of the approaches discussed in the previous section have used structural features of networks and the features of the nodes and edges for link prediction. In friendship networks, such as Facebook, the nodes are users and the links are the relationships between users, whereas in coauthorship networks, the nodes are authors, and the edges are the publications by these authors. In both cases, similarities between nodes, links and the structural features of the networks can use to predict future links. For example, the features such as number of common neighbors of a node pair, Jaccard's coefficient [Manning 08] can be computed or any other feature introduced in past researches can be computed. Once these features are calculated for a particular node pair, we obtain a vector of values referred to as a *feature vector* [Pavlov 07], which may be correlated with the future possible link between that node pair.

In supervised learning approach, we train the learning system with the feature vectors of each node pair to learn a model which can be used to predict the future links. Once we compute the feature vectors for each node pair in a network, we obtain a set of feature vectors for node pairs that are already linked and another set of feature vectors for node pairs that are not linked. The learning system is trained to learn a model using the feature vectors and the model used to predict unlinked node pairs that are to be linked in the future.

3.1 Features used for link prediction

Table 1 lists the details of the features used in the present study. We used two different combinations of features in the proposed machine learning approach for link prediction. Here, we used a set of features used in [Munasinghe 12] with previous algorithm *PropFlow* [Lichtenwalter 10] and new algorithm *T_Flow* introduced in this paper. One set was used as the *PropFlow combination*, and the other set is the *T_Flow combination*, which includes the *T_Flow* introduced herein.

The existing features are described below.

Adamic/Adar [Adamic 03] This measure indicates if a node pair has a common neighbor which is not common to several other nodes, then the similarity of that particular node pair is higher than the node pairs having neighbors that are common to several other nodes. This measure assigns higher weights to common neighbors that are not common to several other nodes.

Common neighbors Number of common neighbors of a node pair.

Jaccard's coefficient [Manning 08] Normalized measure of common neighbors.

Preferential attachment [Newman 01] This measure indicates that new links are more likely to be formed with nodes of higher degree, or nodes that are popular in the network.

In the formulas in Table 1, v_i, v_j, and v_k denote nodes, and $\Gamma(v_i)$ and $\Gamma(v_j)$ denote the sets of neighbors of v_i and v_j, respectively. In the next section, we discuss the new feature called *T_Flow* introduced in this paper.

3.2 PropFlow Algorithm

The *T_Flow* algorithm introduced here is an extension of the *PropFlow* algorithm [Lichtenwalter 10] which was introduced to assign higher scores for node pairs which have

Table 1: Feature Listing

Feature	Formula	PropFlow combination(PFC)	T_Flow combination(TFC)
Adamic/Adar	$\sum_{v_k \epsilon \Gamma(v_i) \bigcap \Gamma(v_j)} \frac{1}{log\|\Gamma(v_k)\|}$	✓	✓
Common neighbors	$\|\Gamma(v_i) \bigcap \Gamma(v_j)\|$	✓	✓
Jaccard's coefficient	$\frac{\|\Gamma(v_i) \bigcap \Gamma(v_j)\|}{\|\Gamma(v_i) \bigcup \Gamma(v_j)\|}$	✓	✓
Preferential attachment	$\|\Gamma(v_i)\|\|\Gamma(v_j)\|$	✓	✓
PropFlow Score		✓	-
T_Flow Score		-	✓

higher information flow between them. The *PropFlow* algorithm is applicable for both directed and undirected networks. This algorithm used restricted random walk with link weights as the transition probabilities and breadth-first search as the searching method. Link weight is considered as number of cooccurrences of a particular node pair. The *PropFlow* link predictor assigns higher scores for the node pairs which have higher information flows. The random walker starts from a particular node and reach the desired node in l steps and stops when reaching the desired node or revisiting any node. In the *PropFlow* algorithm, the information flow from node i to node j is computed as follows;

$$PropFlow_{ij} = NodeInput_i * \frac{w_{ij}}{SumOutput_i} + \sum Flow\ from\ i\ to\ j\ through\ indirect\ paths \quad (1)$$

$NodeInput_i$ is the starting flow at node i and w_{ij} is the weight of the direct link between node i and node j. $SumOutput_i$ is the total of the weights of links between the node i and its neighbors. The random walker can reach node j through all difference paths which connects node i and node j. The flow between two nodes is the sum of the flows of direct path and all indirect paths. from node i to node j as shown in the Equation 1. The total flow is regarded as the *PropFlow score* for the node pair. The idea of *PropFlow* is shown in Figure 1. This is an example of a coauthorahip network. Each node represents an author and p denote the number of publications. For example, if we assume random walker starts from A the flow between node A and D is $\frac{9}{32}$. The nodes A and D are not directly connected but there are two indirect paths from A to D. They are A→B→C→D and A→B→E→D. We have to note that random walker use breadth-first search and it stops when revisiting any node. Then we do not need to think about path A→E→C→D because as soon as random walker revisits C it stops.

3.3 T_Flow Algorithm

The information flow in social networks doesn't depend only on the link weights. The activeness of the links is a vital factor for information flow. The links become weak or deactivate if nodes have not interacted recently with respect to the current time. Despite of their weights, the weakened or deactivated links can cause a decay in information flow. We therefore, introduced an extension of *PropFlow* referred

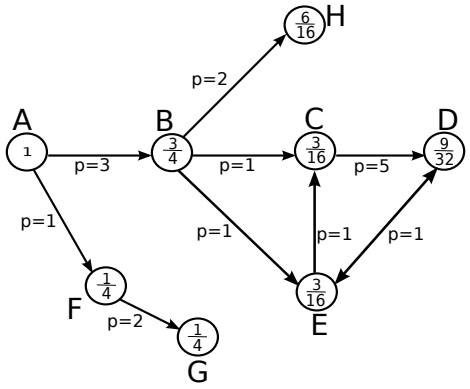

Figure 1: An example of a PropFlow algorithm

to as *T_Flow* in order to consider the effect of active links for information.

The new algorithm called *T_Flow* also use the restricted random walk and breadth-first search as the searching method. The random walker starts from a particular node and reach the desired node in l steps. The transition probabilities are calculated using link weights. The time stamps of the links or interactions are useful in determining the activeness of the links. Hence, we used the most recent time stamps of the interactions between nodes for our computations. We have listed the *T_Flow* procedure in Algorithm 1. In this algorithm, length l defines the neighborhood of a node in terms of number of links. We assumed that decay in information flow as a function of decaying factor α and difference of time stamps of adjacent links. The decaying function is defined as;

$$(1-\alpha)^{|t_x - t_y|} \quad (2)$$

where t_x and t_y are time stamps of two adjacent links. The decaying factor α $(0 < \alpha < 1)$ is the rate of decay per time unit. The value of decaying function become 1 when $\alpha = 0$ which means no decay in information flow and at this point the *T_Flow* is identical to its previous version *PropFlow*. In the T_Flow algorithm, the information flow from node i to node j is computed as follows;

$$T_Flow_{ij} = NodeInput_i * \frac{w_{ij}}{SumOutput_i} * (1-\alpha)^{|t_x - t_y|} + \sum Flow\ from\ i\ to\ j\ through\ indirect\ paths \quad (3)$$

The total flow between two nodes regarded as the *T_Flow score* for the node pair. In Equation 3, t_x is the time stamp of the link which random walker comes into the node i and t_y is the time stamp of direct link between node i and node j. At the start of the random walk, t_x is regarded as the current time and the initial node input is considered as 1.

We have shown the idea of *T_Flow* algorithm with example of coauthorship network in Figure 2. Time stamps of the links are the most recent years of publications and t denote the time and p denote the number of publications. We assumed the current time is 2012. For example, assume that random walker starts from node A. The weight of link AB is 3. Then we need to calculate the information flow f_B for B. In that case, we consider weight and activeness of link AB with respect the current time. Therefore, the information flow of B is;

$$f_B = \frac{3}{1+3} * (1-\alpha)^{|2012-2011|} = \frac{3}{4} * (1-\alpha) \quad (4)$$

There is no decay in information flow because the nodes A and B have interacted in the current year 2011. We calculated the information flow f_D for node D shown in Figure 2 considering link weights of links AB, BC, CD, BE, ED and their time stamps. Thus, the information flow of node D is;

$$f_D = f_c * \frac{5}{5} * (1-\alpha)^{|2009-2007|} + f_E * \frac{1}{1+1} * (1-\alpha)^{|2006-2004|} \quad (5)$$

f_C can be compute as;

$$f_C = f_B * \frac{1}{2+1+1} * (1-\alpha)^{|2011-2007|} = \frac{3}{4} * (1-\alpha) * \frac{1}{4} * (1-\alpha)^{|2011-2007|} \quad (6)$$

f_E can be compute as;

$$f_E = f_B * \frac{1}{2+1+1} * (1-\alpha)^{|2011-2004|} = \frac{3}{4} * (1-\alpha) * \frac{1}{4} * (1-\alpha)^{|2011-2004|} \quad (7)$$

Therefore, the f_D is;

$$f_D = \frac{3}{16} * (1-\alpha)^7 + \frac{3}{32} * (1-\alpha)^{10} \quad (8)$$

We have to note that random walker use breadth-first search and it stops when revisiting any node. Then we do not need to think about path A→E→C→D because as soon as random walker revisits C it stops. The link BC has the time stamp (2007) and the link BE has the time stamp (2004) of three out going links of node B. Therefore, BC is the most active link. Thus, more information should flow through BC than BE which has the same weight as BC but less active than BC.

The *T_Flow* algorithm can compute *T_Flow score* for node pairs in a network provided with link weights and time stamps of the links.

Algorithm 1: T_Flow Predictor

Data: network $G = (V, E)$, node v_s, length l, decaying factor α
Result: Score S_t for all neighbors of v_s within l-length neighborhood

```
begin
    insert v_s into Found
    push v_s into NewSearch
    push CurrentTime into Time
    insert (v_s,1) into Scores
    for Distance← 0 to d do
        OldSearch ⟵ NewSearch
        empty NewSearch
        while OldSearch is not empty do
            pop v_i from OldSearch
            pop t_x from Time
            find NodeInput using v_i in Scores
            SumOutput ← 0
            t_y ← 0
            for each v_j in l-length neighborhood of v_i do
                add weight of e_ij to SumOutput
            end
            Flow ← 0
            for each v_j in l-length neighborhood of v_i do
                w_ij ← weight of e_ij
                t_y ← t_ij
                Flow ←
                NodeInput * (w_ij / SumOutput) * (1-α)^|t_x - t_y|
                insert or sum (v_i, Flow) into Scores
            end
            if v_i is not in Found then
                insert v_i into Found
                push v_i into NewSearch
                push t_y into Time
            end
        end
    end
end
```

4. Experimental evaluation

We test the effectiveness of *T_Flow* algorithm for link prediction using two real-world social network data sets. The first data set is a piece of Facebook friendship network data collected from the regional Facebook network of New Orleans [Viswanath 09]. This data set consist of 60,290 users who are connected by 1,545,686 links. The second data set is a coauthorship data set extracted from 87,413 publications on condensed matter physics from 1997 to 2007 in the *cond-mat archive*[*1]. Both data sets are undirected networks. In the Facebook data set, link weight represents the number of wall postings between two users. In the coauthorship data, the link weight was calculated using method introduced in [Newman 01] which is explained here. Let i and j are two authors and δ_i^k and δ_j^k are indicator functions. If author i is a coauthor of paper k then $\delta_i^k = 1$ and zero otherwise. If paper k has n_k authors, the weight of collaboration w_{ij} between two authors i and j is computed

∗1 http://arxiv.org/archive/cond-mat/

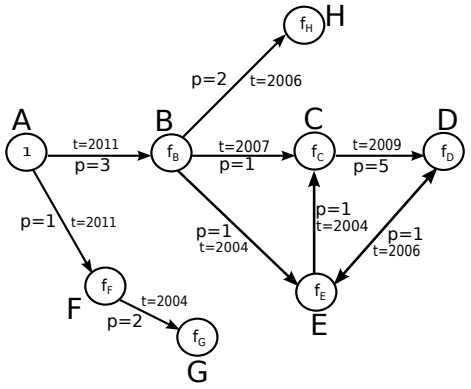

Figure 2: An example of a T_Flow algorithm

as the summation of all shared papres;

$$w_{ij} = \sum_k \frac{\delta_i^k \delta_j^k}{n_k - 1} \tag{9}$$

In both networks, the time stamps of the links are the time stamps of the most recent interactions between nodes. Wall postings considered as the interactions between users in Facebook data and publications considered as the interactions between authors in coauthorahip data.

In our experiments, J48 Weka implementation [Hall 09] of C4.5 decision tree algorithm [Quinlan 93] was used with 10-fold cross validation. Both network data sets used in the experiments are very sparse so, we used SMOT oversampling algorithm [Chawla 02] in order to dealt with class imbalance problem. The activeness of links in rapidly evolving networks such as Facebook depends on temporal factors. Some of them are special events happen during a certain period of time, change of location of the users. In such kind of situations, we need to use data collected over that particular period to study the affect of that particular temporal factors for activeness of links. Therefore, we used different data sets for each real world networks. We train the decision tree algorithm for Facebook data using wall postings in two consecutive weeks to predict links in the following week. We repeated the experiment for six data sets extracted within six months period of facebook data using 10-fold cross validation for each data set and computed the average performance. For the coauthorahip data, we train the decision tree algorithm using five consecutive years of coauthor data to predict links in the following year. We repeated the experiment for six data sets extracted within ten years period of the coauthorship data using 10-fold cross validation for each data set and computed the average performance. For both data sets, we restricted the neighborhood of a node to three which means we excluded the nodes that are three links away from a node. The unit of time for Facebook data is days and for the coauthorship data it is years. Precision, recall and F-measure are used as performance metrics of our experiments. Two differnt feature combinations used in the experiments. The *PropFlow combination* includes *PropFlow* algorithm and *T_Flow combination* includes *T_Flow* algorithm. At first, we tested the

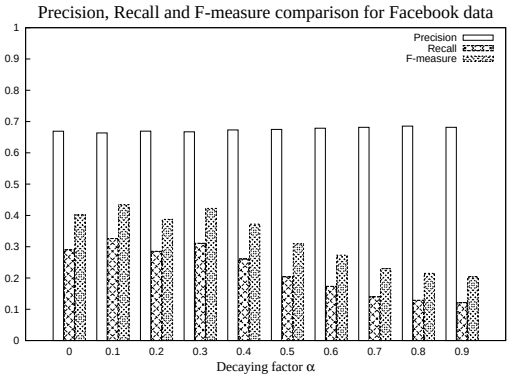

Figure 3: Performance of *T_Flow combination* for different α values (Facebook data)

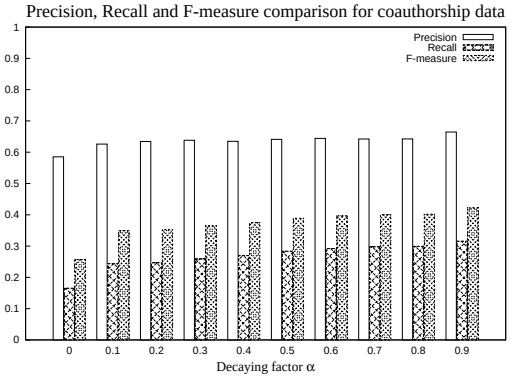

Figure 4: Performance of *T_Flow combination* for different α values (coauthorship data)

performance of *T_Flow combination* by varying α from 0.1 to 0.9 in order to investigate α value which gives the better results for *T_Flow combination* . Then we compared *T_Flow combination* with *PropFlow combination*.

4.1 Experimental results

In this section we have present the experimental results obtained for each data set. The decaying factor α could depends on the network and time unit of time. So, we investigated the parameter value first. Figure 3 show the performances of *T_Flow combination* for different values of α for Facebook data. The F-measure decrease as decay increase as shown in Figure 3. We used two weeks data to predict links in the following week and computed the average of performance metrics for six data sets. The time unit is days and its range vary from 0 to 14. Therefore, the range of decaying function could be large even for smaller α value and as a consequence the results are better for the smaller α values as shown in Figure 3 for Facebook data set.

Figure 4 show the of performances of *T_Flow combination* different with values of α for coauthorship data. The F-measure increase as decay increase as shown in Figure 4. The activeness of links in coauthorahip networks are not change rapidly as authors work together for long time to write research papers. For the experiments with coautho-

Table 2: Comparison of *PropFlow combination* and *T_Flow combination* for Facebook data. ($\alpha = 0.1$)

	Precision	Recall	F-measure
PropFlow	0.6692	0.2898	0.4023
T_Flow	0.6637	0.3260	0.4342

Table 3: Comparison of *PropFlow combination* and *T_Flow combination* for coauthorahip data. ($\alpha = 0.9$)

	Precision	Recall	F-measure
PropFlow	0.5852	0.1655	0.2567
T_Flow	0.6647	0.3157	0.4218

rahip data, we used publication data of six years, five years data to predict links in the following year. Here, the time unit is years and its range is 0 to 5 which is relatively small. Larger α value is required to obtain wide range of values for decaying function and as a consequence the results are better for the larger α values as shown in Figure 4 for coauthorahip data set.

The highest F-measure for Facebook data was obtained at $\alpha = 0.1$. We summarized the comparison of *T_Flow combination* with *PropFlow combination* in the Table 2 for Facebook data when $\alpha = 0.1$. When the decay in information flow per day is 10% the F-measure for *PropFlow combination* is 0.4023 and F-measure for *T_Flow combination* is 0.4342. It is a 3% improvement by the new method *T_Flow combination*. The highest F-measure for coauthorahip data was obtained at $\alpha = 0.9$. We summarized the comparison of *T_Flow combination* with *PropFlow combination* in the Table 3 for coauthorahip data when $\alpha = 0.9$. When the decay in information flow per year is 90% the F-measure for *PropFlow combination* is 0.2567 and F-measure for *T_Flow combination* is 0.4218. It is 17% improvement by the new method *T_Flow combination*. This is a significant improvement according to the t-test with 5% significance level. We have to note that the results at $\alpha = 0$ corresponds to the *PropFlow combination* which doesn't consider the decay of information flow. In this study, we used only wall postings as user interactions for Facebook data. However, wall posting is not the only mean of interactions. The users can interact in many other ways such as photo tagging, liking, etc. On the other hand, the evolution of coauthorship networks depends on many other factors except coauthouring papers. For example, researchers could change research fields according to emerging research trends and make associations with new researchers. The use of such data can increase effectiveness of *T_Flow* algorithm. We will consider those factors in our future works. Besides that, we will investigate methods to estimate α for different kind of social networks and time units.

5. Conclusion

In this paper, we introduced new algorithm *T_Flow* based on information flow which can use for link prediction in social networks. *T_Flow* algorithm assigns scores for the node pairs according to the information flow. The main characteristic of *T_Flow* algorithm is that it considers the impact of activeness of the links for information which has not been discussed in the previous method. We combined the activeness of the links and link weights in *T_Flow* algorithm and investigated how it affect the information flow which is a vital factor for link evolution. The experimental results shows that *T_Flow* algorithm outperform the previous *PropFlow* algorithm which considers only the impact of link weights for information flow. Thus, *T_Flow* algorithm is better for link prediction in social networks where the the activeness of the link vary over time.

References

[Adamic 03] Lada A. Adamic and Eytan Adar. Friends and neighbors on the web. *Social Networks*, 25:211–230, 2003.

[Backstrom 11] Lars Backstrom and Jure Leskovec. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the Forth International Conference on Web Search and Web Data Mining*, pages 635–644, 2011.

[Chawla 02] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[Hasan 06] Mohammad Al Hasan, Vineet Chaoji, Saeed Salem, and Mohammed Zaki . Link prediction using supervised learning. In *Proceedings of SDM 06 workshop on Link Analysis, Counterterrorism and Security*, 2006.

[Hall 09] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, 2009.

[Kahanda 09] Indika Kahanda and Jennifer Neville. Using transactional information to predict link strength in online social networks. In *Proceedings of the Third International Conference on Weblogs and Social Media*, 2009.

[Kashima 06] Hisashi Kashima and Naoki Abe. A parameterized probabilistic model of network evolution for supervised link prediction. In *Proceedings of the 6th International Conference on Data Mining*, pages 340–349, 2006.

[Kashima 09] Hisashi Kashima, Tsuyoshi Kato, Yoshihiro Yamanishi, Masashi Sugiyama, and Koji Tsuda. Link propagation: A fast semi-supervised learning algorithm for link prediction. In *Proceedings of the SIAM International Conference on Data Mining*, pages 1099–1110, 2009.

[Liben-Nowell 03] David Liben-Nowell and Jon Kleinberg. The link prediction problem for social networks. In

Proceedings of the 12th International Conference on Information and Knowledge Management, pages 556–559, 2003.

[Lichtenwalter 10] Ryan N. Lichtenwalter, Jake T. Lussier, and Nitesh V. Chawla. New perspectives and methods in link prediction. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 243–252, 2010.

[Manning 08] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[Munasinghe 12] Lankeshwara Munasinghe and Ryutaro Ichise. Time score: A new feature for link prediction in social networks. *IEICE Transactions on Information and Systems*, E95-D:pages 821–828, 2012.

[Newman 01] M. E. J. Newman. Clustering and preferential attachment in growing networks. *Phys. Rev. E*, 64(2):025102, Jul 2001.

[Newman 01] M. E. J. Newman. Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality. *Phys. Rev. E*, 64(1):016132, Jun 2001.

[Pavlov 07] Milen Pavlov and Ryutaro Ichise. Finding experts by link prediction in co-authorship networks. In *Proceedings of the Workshop on Finding Experts on the Web with Semantics*, pages 42–55, November 2007.

[Quinlan 93] J. Ross Quinlan. *C4.5: programs for machine learning*. 1993.

[Sachan 10] Mrimaya Sachan and Ryutaro Ichise. Using abstract information and community alignment information for link prediction. 2(4):334–339, 2010.

[Scripps 08] Jerry Scripps, Pang-Ning Tan, Feilong Chen, and Abdol-Hossein Esfahanian. A matrix alignment approach for link prediction. In *Proceedings of the 19th International Conference on Pattern Mining*, pages 1–4, 2008.

[Sharan 08] Umang Sharan and Jennifer Neville. Temporal-relational classifiers for prediction in evolving domains. In *Proceedings of the Eighth IEEE International Conference on Data Mining*, pages 540–549, 2008.

[Viswanath 09] Bimal Viswanath, Alan Mislove, Meeyoung Cha, and Krishna P. Gummadi. On the Evolution of User Interaction in Facebook. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Social Networks*, , August 2009.

[Wang 07] Chao Wang, Venu Satuluri, and Srinivasan Parthasarathy. Local probabilistic models for link prediction. In *Proceedings of the 7th IEEE International Conference on Data Mining*, pages 322–331, 2007.

[Wohlfarth 08] Till Wohlfarth and Ryutaro Ichise. Semantic and event-based approach for link prediction. In *Proceedings of the 7th International Conference on Practical Aspects of Knowledge Management*, pages 50–61, 2008.