Link Prediction in a Modified Heterogeneous Bibliographic Network

John Boaz Lee*†, Henry Adorna†
*Department of Information Systems and Computer Science, Ateneo de Manila University
†Algorithms and Complexity Lab, Department of Computer Science, University of the Philippines - Diliman
jtlee4@up.edu.ph, hnadorna@dcs.upd.edu.ph

***Abstract*—Researchers have discovered, in recent years, the advantages of modeling complex systems using heterogeneous information networks. These networks are comprised of heterogeneous sets of nodes and edges that better represent the different entities and relationships often found in the real world. Although heterogeneous networks provide a richer semantic view of the data, the added complexity makes it difficult to directly apply existing techniques that work well on homogeneous networks.**

In this paper, we propose a graph modification process that alters an existing heterogeneous bibliographic network into another network, with the purpose of highlighting the important relations in the bibliographic network. Several importance scores, some adopted from existing work and others defined in this work, are then used to measure the importance of links in the modified network. The link prediction problem is studied on the modified network by implementing a random walk-based algorithm on the network. The importance scores and the structure of the modified graph are used to guide a random walker towards relevant parts of the graph, *i.e.* towards nodes to which new links will be created in the future. The different properties of the proposed algorithm are evaluated experimentally on a real world bibliographic network, the DBLP. Results show that the proposed method outperforms the state-of-the-art supervised technique as well as various approaches based on topology and node attributes.

***Index Terms*—heterogeneous information network, random walk with restart, link prediction, relative importance**

I. Introduction

One of the fundamental problems in network analysis is predicting the emergence of new links. It is a problem with practical applications in many different domains. In biology, for instance, experimental methods for identifying protein-protein interaction are often costly and researchers have begun to explore *in silico* methods to help predict these interactions [2], [8]. In online social networks like Facebook, value can be added to the service by accurately recommending new connections to users [3].

Most of the early works [1], [11] in link prediction focused mainly on the definition of similarity measures based on structural information encoded in the network. These kind of link prediction methods aimed at inferring future connections by analyzing the structural patterns of the current network. The idea was that new links could be expected to appear between nodes that were most similar with regard to network structure. One of the simplest structure-based measure is the common neighbor count. Nodes that shared a large set of common neighbors are deemed to be most similar.

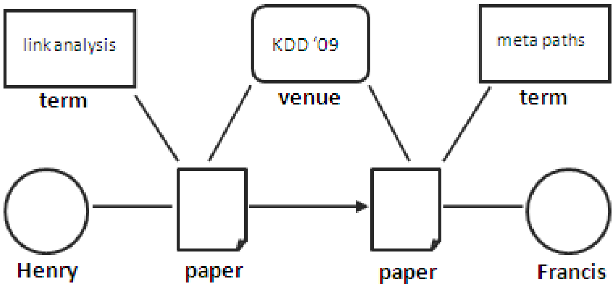

Fig. 1. An example of a heterogeneous bibliographic network.

Many methods based on supervised learning were subsequently introduced in the literature [3], [7]. These methods work by learning the coefficients or weights corresponding to selected features in the training data.

A notable example of a supervised link prediction algorithm can be found in a recent work by Backstrom and Leskovec [3]. The algorithm in [3] learns an edge weight function such that a random walk on the network is more likely to visit positive training examples in the weighted graph.

Probabilistic methods that aim to build models to capture the correlations among links have also been explored [9], [10]. A recent work studied the problem in an environment where existing links in the network are not fully observed [10]. In this work, the graph was modeled in a probabilistic manner.

Most of the work in link prediction [5], [6], including all the above-mentioned ones, deal with homogeneous information networks wherein nodes and edges are of a single type. It is more natural, however, to model real world networks using a heterogeneous set of objects and relationships. For example, a co-authorship network, which is comprised of just author nodes linked together by co-authorship relations, can be further refined by adding *papers*, *venues*, and *topics* to create a *heterogeneous bibliographic network*.

Fig. 1 shows an example of a heterogeneous bibliographic network with the various relations that exist among the different types of nodes: (1) authors and papers are connected by “write” and “written by” relations, (2) papers and topics are linked by “contain” and “contained in” relations, (3) an edge between venues and papers signify the "publish" and "published in" relations, and (4) papers can be connected to one another by the "cite" and "cited by" relations. It is clear that this heterogeneous set of nodes and links provide a semantically richer view of the data. We can now tell that the authors Henry and Francis (from Fig. 1) both published papers in the same venue and the former's paper cited the latter's paper. This kind of information is not found intuitively in a co-authorship network.

In a recent work [14], researchers have shown that by considering the different relations captured by *meta paths*, paths comprised of different types of nodes and links, link prediction in a heterogeneous bibliographic network can be improved. In this work, we modify the network to make it more suitable for random walks - which, to some extent, is more natural for measuring node similarity since the entire topology is taken into account. We make sure to capture, in our new graph, the important relations identified in the previous work [14]. Importance measures are then defined to guide the random walker along the graph. Our contributions are:

- A link prediction model based on random walks is developed for heterogeneous bibliographic networks. The random walk algorithm runs on a modified heterogeneous bibliographic network to highlight important relations.
- We study the effect of several importance measures in link prediction, this helps us understand the mechanisms behind co-author relationship building.
- Experiments on the real world DBLP network demonstrates the efficiency of the proposed method.

II. The Proposed Graph Modification Process and Some Definitions

We introduce in this section the definition of a heterogeneous bibliographic network, our proposed graph modification process, and the link prediction task in the context of the modified graph.

A. Heterogeneous Bibliographic Network Based on DBLP

We base our bibliographic network on the real world DBLP network. The DBLP bibliographic dataset, as provided by [17], contains publication information such as paper title, authors, venue (journal or conference), citations, and publication date. We use the sequential topic mining algorithm in [12] to identify popular phrases from paper titles which we use as topics.

Our network can then be defined as a directed graph $G = \langle V, E\rangle$, comprised of a vertex set V and an edge set E, that has a type mapping function $\tau : V \rightarrow \mathcal{T}$ and a relation mapping function $\rho : E \rightarrow \mathcal{R}$. Each node $v \in V$ can be mapped to a particular type $\tau(v) \in \mathcal{T}$, for $\mathcal{T} = \{\text{author, paper, venue, topic}\}$, and each link $e \in E$ denotes a certain relation $\rho(e) \in \mathcal{R}$, for $\mathcal{R} = \{\text{write, written by, contain, contained in, cite, cited by, publish, published in}\}$. Furthermore, each paper node in $V_p \subset V$ can be mapped by a function $\phi : V_p \rightarrow \mathbb{N}$ to the year it was published. Fig. 2 shows the schema of the DBLP network.

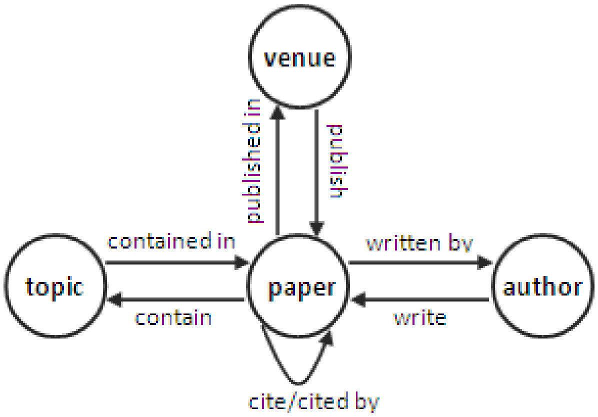

Fig. 2. Schema of the DBLP Bibliographic Network.

B. Proposed Graph Modification Process

We now describe our proposed graph modification process. To construct the new network, we create a graph resembling an "attribute" graph [13]. In particular, we build upon the work in [19], which describes a method to construct a bi-typed network comprised of person nodes and attribute nodes. In our work, we introduce additional types of nodes and edges that will allow the network to better capture the important relations present in a bibliographic network.

Given a bibliographic network $G = \langle V, E\rangle$ where $V = V_a \cup V_p \cup V_v \cup V_t$, V_a is the set of author nodes, V_p is the set of paper nodes, V_v is the set of venue nodes, and V_t is the set of topic nodes; we create an undirected graph $G' = \langle V', E', W\rangle$ based on the relations found in G. Here $W : E' \rightarrow \mathbb{R}^+$ is a weight mapping from the new set of edges E' to $\mathbb{R}^+$.

Following [19], we create an author node in G' corresponding to each author in G and refer to these as *author nodes*. We also create a node for each topic, cited paper, and venue and call these *attribute nodes*. Therefore $V' = V_{auth} \cup V_{attri}$, where $V_{auth} = V_a$ is the set of author nodes and $V_{attri} = V_t \cup V_v \cup V_c$ is the set of attribute nodes, $V_c \subset V_p$ is the set of papers that were cited by other papers (note that only papers that have been cited are included as attribute nodes in V_{attri}).

We now define the different types of edges that are included in the new network. These are:

1. Structural links. For two authors a, $a' \in V_{auth}$, $\langle a, a'\rangle \in E'$ if $\exists p \in V_p$ s.t. $\langle a, p\rangle \in E$ and $\langle a', p\rangle \in E$. In other words, we add an edge between two author nodes a and a' in V' if they are co-authors.

2. Attribute links. For two nodes x, $y \in V'$, $\langle x, y\rangle \in E'$ if

- $x \in V_{auth}$, $y \in V_t$, and $\exists p \in V_p$ s.t. $\langle x, p\rangle \in E$ and $\langle p, y\rangle \in E$.
- $x \in V_{auth}$, $y \in V_v$, and $\exists p \in V_p$ s.t. $\langle x, p\rangle \in E$ and $\langle p, y\rangle \in E$.
- $x \in V_{auth}$, $y \in V_c$, and $\exists p \in V_p$ s.t. $\langle x, p\rangle \in E$ and $\langle p, y\rangle \in E$.

That is, an author node is connected to a topic node in the new graph if a paper authored by the corresponding author

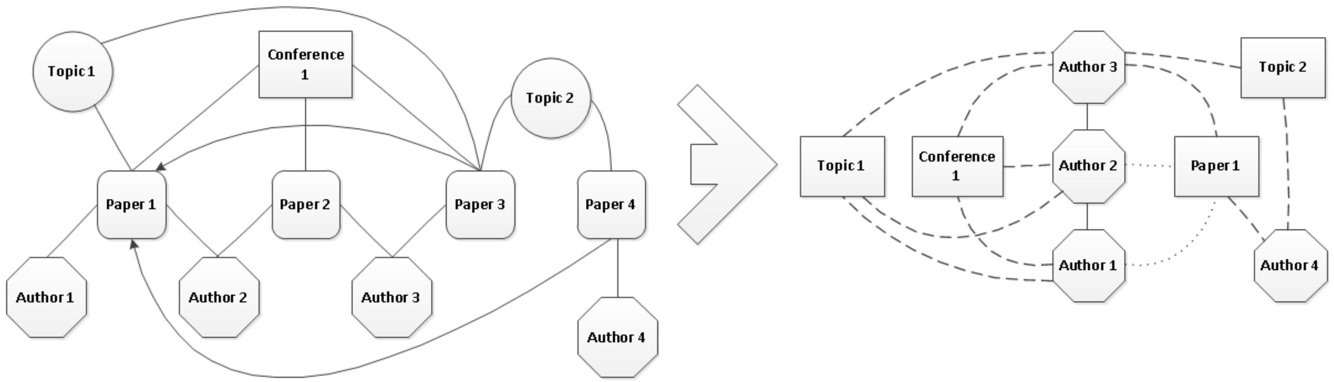

Fig. 3. An example of the original bibliographic network is shown on the left while the modified network is shown on the right. In the new network, the topics, conferences, and cited papers are now attribute nodes and attribute links (dashed lines) connect them to authors. The solid lines represent structural links while the dotted lines denote trivial self-citation links - this last type of link helps connect authors to others who have cited their work and vice versa.

contained the topic. Similarly, an author node is linked to a cited paper node (or venue node) if a paper written by the corresponding author cited the paper (or was published in the venue). Note that it is straightforward to treat topics found in an author's paper as attributes by linking the corresponding author and topic nodes. Venues that publish an author's work can also be treated as the author's attributes by linking the corresponding author and venue nodes. However, it is not immediately clear how we can represent paper citations found in the original bibliographic network G in the modified network G'. A clever way to incorporate citation information in the modified network is to treat these as attributes, thus cited papers are included as attribute nodes in G'; *i.e.* author nodes are connected to cited paper nodes if the corresponding authors cited the corresponding papers. To go one step further, one can choose to create attribute nodes corresponding to papers that have cited other papers ("citing papers"); and author nodes are linked to the same "citing paper" node if their works were cited together by the citing paper. We do not include this last type of node, however, in our study.

3. Trivial self-citation links. For an author $a \in V_{auth}$ and a cited paper $c \in V_c$, $\langle a, c\rangle \in E'$ if $\langle a, c\rangle \in E$. Note that this edge definition is redundant if $\exists p \in V_p$ s.t. $\langle a, p\rangle \in E$ and $\langle p, c\rangle \in E$ meaning the author a has cited the paper c that he has written from another paper p that he also authored. These links connect author nodes to cited paper nodes if the corresponding author wrote the corresponding cited paper. Note that trivial self-citation links act like attribute links since they connect author nodes with attribute nodes.

Fig. 3 shows an example of a bibliographic network before and after it undergoes our proposed graph modification process. The new graph is constructed in this way to capture the important relations between authors that contribute to co-authorship relation building as discussed in [14]. The structural links capture the idea that people who share co-authors should be closer to each other in the network and are thus more likely to co-author in the future. The attribute links help connect authors who: (1) publish in the same venue, (2) write about the same topics, and (3) cite the same papers, as these factors also increase the chance of co-authorship. Finally, the trivial self-citation links are added onto the new graph for two reasons. They help to link an author to those that have cited his works, and conversely, they connect the author to the authors whose works he has cited. All these help capture the important relations behind co-authorship link building [14].

C. *Link Prediction in Modified Graph*

Given the modified network $G' = \langle V', E', W\rangle$, an author node $a \in V_{auth}$, and $k \in \mathbb{N}$, the link prediction task aims to identify the nodes $a' \in V_{auth}$ that a creates structural links to in the future. Specifically, if G' represents the bibliographic network at some time t, the link prediction task tries to infer the set of authors $\{a'_1, ..., a'_k\} \subset V_{auth}$ that a is most likely to co-author with in some future time interval t'.

III. Edge Weighting Scheme

A. *Importance Measures*

We now describe several importance measures that help capture the concept of the importance of a node in the modified network. These measures are used to bias the weights of links. The first two measures discussed here are from [19]. The latter two measures are introduced in this work.

1. Global Importance Measure. The global importance measure of an attribute node x measures the percentage of actual co-authors over all possible co-authors among authors who are linked to the attribute node x.

$$g(x) = \frac{\sum_{\langle a,a'\rangle \in E'} e_x(a, a')}{\binom{n_x}{2}}$$

Here, n_x is the degree of x, or simply the number of author nodes linked to it. $e_x(a, a') = 1$ if co-authors a and a' are both linked to attribute x, and is zero otherwise.

We can see that this measure is helpful because it rewards attributes that are specific while penalizing general attributes. For instance, a general computer science conference attracts researchers working in very different fields who do not have much in common and are thus less inclined to co-author with one another. On the contrary, a highly specialized conference like ACM SIGKDD attracts authors – many of whom have co-authored in the past – that specialize in data mining and the chances of two authors co-authoring here should be greater.

2. Local Importance Measure. The local importance $l_a(x)$ of a node $x \in V'$ relative to an author node a is defined as

$$l_a(x) = \sum_{a' \in N_{auth}(a)} A(a', x)$$

where $N_{auth}(a)$ is the set of a's co-authors and $A(a', x) = 1$ if a' is linked to the node x, $A(a', x) = 0$ otherwise. Local importance captures the importance of a node in the graph relative to the author's friends or co-authors.

The local importance measure is also important because it looks at node importance from an author's co-authors' perspective. To see why this is relevant, we give an example. An author specializing in membrane computing authors two papers. One is published in a highly specialized conference for natural computing where many of the authors past co-authors have also submitted their work. The other paper is the product of a chance encounter between the author and an individual who specializes in AI. Their paper is published in a conference specializing in AI. Now both conferences, being equally specialized, happen to have the same global importance. However, the local importance for the first conference should be greater than that of the latter since the conference on natural computing is where most of the author's co-authors can also be found (revealing the "real interest" of the author).

3. Frequency Importance Measure. Aside from the first two measures, we would also like to measure how frequent an interaction occurs. For example, the link between two co-authors who have co-authored twenty papers in the past should be stronger than that between co-authors who co-authored a single paper. The same can be said for a link to a topic that has been mentioned multiple times in the past versus another that was only mentioned once.

The frequency measure $f_a(x)$ of a node $x \in V'$ relative to an author a is

$$f_a(x) = k^{freq(a,x)}$$

where $freq(a, x) = |P_a \cap P_x|$, $P_a = \{p \in V_p : \langle p, a\rangle \in E\}$ is the set of paper nodes linking to author node a, and $P_x = \{p \in V_p : \langle p, x\rangle \in E\}$ is the set of paper nodes linking to node x in G. For an author node a, P_a is the set of papers authored by a. For a node x, say a cited paper, P_x is then the set of papers citing x, $|P_a \cap P_x|$ is then the number of times a cites x. Here, we set $k = 1.1$ since this yielded better performance compared to higher values; k should be in $(1, \infty]$.

Note that $freq(a, x) = 0$ for a cited paper x connected to an author node a by a trivial self-citation link since a never explicitly cited x. For this special case, we set $freq(a, x) = 1$.

4. Recency Importance Measure. Finally, we calculate the importance of a link by the recency of interaction. Through this measure, we are able to strengthen links with recent activity. Intuitively, one is more likely to find co-authors among the friends of someone whom one has recently worked with versus someone with whom one has had no collaboration with for the last twenty years. Our recency measure $r_a(x)$ for a node $x \in V'$ with regard to author node a is defined as follows.

$$r_a(x) = \frac{1}{k^{ly - rec(a,x)}}$$

We denote by ly the year of the most recently published paper in the dataset and $rec(a, x) = \arg\max_{p \in P_a \cap P_x} \phi(p)$ is the year in which the latest paper between author a and the node x was published in the original network G. Recall that the function ϕ maps a paper to the date it was published.

Note that $rec(a, x)$ is undefined for a cited paper node x connected to an author node a through a trivial self-citation link. In this case, we set $rec(a, x) = \phi(x)$, which is simply the year the paper x was published.

B. Weighting Links According to Relative Importance

We now discuss how to assign weights to the links based on the measures defined. We start by defining the aggregate relative importance scores of nodes linked to an author node.

Given an author node $a \in V_{auth}$, an attribute node $x \in V_{attri}$ connected to a, and another author node $a' \in V_{auth}$ who is a co-author of a, the relative importance score $w_a(\cdot)$ of a node relative to a can then be defined as follows.

$$w_a(x) = \alpha g(x) + \beta l_a(x) + \gamma f_a(x) + \delta r_a(x)$$

$$w_a(a') = (\frac{\alpha}{3} + \beta) l_a(a') + (\frac{\alpha}{3} + \gamma) f_a(a') + (\frac{\alpha}{3} + \delta) r_a(a')$$

The parameters $\alpha, \beta, \gamma, \delta \in [0, 1]$ control how much a measure contributes to the final score, $\alpha + \beta + \gamma + \delta$ = 1.

Note that the importance score of an attribute with regard to an author is a combination of all four importance measures. In other words, an attribute is important to an author a if it is (1) linked to a subset of author nodes that have a high clustering coefficient, (2) linked to many of a's co-authors, (3) often associated with a, and (4) recently associated with a.

On the other hand, the importance of a co-author a' is only dependent on three measures since we are only concerned with whether a has had recent and frequent interactions with a' and whether a's other co-authors are also linked to a'. Whether the co-author a' is a generalist or a specialist does not matter.

The normalized edge weight $W(a, \cdot)$ of the edge $\langle a, \cdot\rangle$ from an author node a can now be defined as follows.

$$W(a, x) = \begin{cases} \frac{\lambda w_a(x)}{\sum_{x' \in N_{attri}(a)} w_a(x')} & \text{: if } |N_{attri}(a)| > 0 \text{ and} \\ & \quad |N_{auth}(a)| > 0; \\ \frac{w_a(x)}{\sum_{x' \in N_{attri}(a)} w_a(x')} & \text{: if } |N_{attri}(a)| > 0 \text{ and} \\ & \quad |N_{auth}(a)| = 0; \\ 0 & \text{: otherwise.} \end{cases}$$

$$W(a, a') = \begin{cases} \frac{(1-\lambda) w_a(a')}{\sum_{\hat{a} \in N_{auth}(a)} w_a(\hat{a})} & \text{: if } |N_{auth}(a)| > 0 \text{ and} \\ & \quad |N_{attri}(a)| > 0; \\ \frac{w_a(a')}{\sum_{\hat{a} \in N_{auth}(a)} w_a(\hat{a})} & \text{: if } |N_{auth}(a)| > 0 \text{ and} \\ & \quad |N_{attri}(a)| = 0; \\ 0 & \text{: otherwise.} \end{cases}$$

The parameter λ controls how much we depend on a certain type of link. $N_{attri}(a)$ is the set of attribute nodes connected to author a, while $N_{auth}(a)$ is the set of author nodes connected to a. Please note that in all the above calculations, we treat trivial self-citation links as attribute links.

We now define the edge weight $W(x, a)$ of a link from an attribute node x to an author node a.

$$W(x,a) = \frac{(\frac{\alpha}{2}+\frac{\beta}{2}+\gamma)f_a(x)+(\frac{\alpha}{2}+\frac{\beta}{2}+\delta)r_a(x)}{\sum_{a'\in N_{auth}(x)}(\frac{\alpha}{2}+\frac{\beta}{2}+\gamma)f_{a'}(x)+(\frac{\alpha}{2}+\frac{\beta}{2}+\delta)r_{a'}(x)}$$

for $N_{auth}(x)$ is the set of author nodes connected to attribute node x. This simply biases the weights towards authors who are frequently and recently associated with the attribute.

Note that our edge weighting scheme biases the weights of all edges in the modified network by using a combination of the different importance measures. This allows us to strengthen or weaken the meta paths between the author nodes in the graph. In this way, we are able to bring similar authors closer together through paths with stronger cumulative edge weights.

IV. Random Walk Link Prediction Algorithm

To perform link prediction in our context, an algorithm based on random walks is defined on the new graph. A random walk process is started from a single query author node, and the stationary random walk probabilities to the other nodes on the graph is then considered as the *link relevance*, which is the likelihood of a link occurring between the author node and the respective nodes. Since the edges in the graph were weighted, the random walk starting from an author a is more likely to discover nodes that are linked to a through "important" paths.

To calculate the link relevance of nodes with regard to a particular query author a^*, we define the random walk process on the new graph as follows

$$\begin{aligned} r_a^{\langle t\rangle} &= (1-c)\sum_{a'\in N_{auth}(a)} W(a',a)r_{a'}^{\langle t-1\rangle} \\ &\quad + (1-c)\sum_{x'\in N_{attri}(a)} W(x',a)r_{x'}^{\langle t-1\rangle} + cr_a^{\langle 0\rangle} \\ r_x^{\langle t\rangle} &= (1-c)\sum_{a'\in N_{auth}(x)} W(a',x)r_{a'}^{\langle t-1\rangle} \end{aligned}$$

where $r_a^{\langle t\rangle}$ is the random walk probability from author a^* to author a after the t^{th} iteration, and $r_x^{\langle t\rangle}$ is the random walk probability from a^* to the attribute x at time t. The random walk vector r at time zero has all its components initialized to zero except $r_a^{\langle 0\rangle} = 1$ if $a = a^*$. The parameter c is the restart probability, it controls how often we should restart the random walk at a^*, the higher the value for c the more restricted the random walk is to the local neighborhood since the random walker is more likely to keep restarting at node a^*. When $c = 0$, the random walker starts at a^* and explores the graph until the process has converged.

A summary of the proposed method is described below.

Algorithm 1 Random Walk Link Prediction Algorithm

1. Input: A heterogeneous bibliographic network $G = \langle V, E\rangle$, an author $a^* \in V$, and the parameters λ, α, β, γ, δ, and c.

2. Create the new graph $G' = \langle V', E', W\rangle$ based on the graph modification process described in section 2.

3. Assign weights to the edges in G' using the global importance, local importance, frequency, and recency measures weighted according to λ, α, β, γ, and δ.

4. Create the random walk vector r and set $r_a^{\langle 0\rangle} = 1$ if $a^* = a$ and zero otherwise. Set $r_x^{\langle 0\rangle} = 0$ for all attribute nodes $x \in V_{attri}$. Iterate to update r until values have converged.

5. Select all author nodes a' where $a' \notin N_{auth}(a^*)$. Order the selected authors by decreasing $r_{a'}^{\langle *\rangle}$, where $r_{a'}^{\langle *\rangle}$ is the stationary random walk probability to a'.

6. Output the ordered list as the recommended potential co-authors.

V. Experimental Setup, Results and Discussion

A. Dataset

We use the real world DBLP bibliographic dataset to create our network, authors of papers published in the World Wide Web (WWW) conferences from years 2001 to 2008 are considered as the author nodes in our network. For each of the 2,505 authors, we get their publication history from 1991 to 2007. We use this information to build the modified graph which contains a total of 76,814 topics, 52,136 cited papers, and 7,738 venues. The attribute nodes with degree one were not counted. The authors in the modified graph were connected to an average of 8.39 co-authors and made an average of 3.35 new co-author links in the succeeding time period $[2008, 2010]$. Due to hardware constraints, the sampled dataset is only a subset of the DBLP. However, this is already a good sample of the larger network as the WWW conferences attracts researchers from varied fields [16]. Moreover, this dataset is similar to the one used by [19] in their tests.

B. Test Setup

The proposed algorithm is run on the modified network for each author with at least one co-author and the top-k candidate co-authors are predicted. The precision and recall of the algorithm is used to benchmark the algorithm. Prec@k $= \frac{1}{|T|}\sum_{o\in T}\frac{|P_k(o)|}{k}$ where T is the training set and $P_k(o)$ is the set of authors in the list of top-k candidates that o truly created a link to in the future. In other words, precision measures the correct predictions among the top-k predicted co-authors. Rec@k $= \frac{1}{|T|}\sum_{o\in T}\frac{|P_k(o)|}{|R(o)|}$ where $R(o)$ is the set of all authors that o linked to in the future.

We also tested several benchmark models. The first set of methods are based on structural properties. These are:

- ComNeigh: $score(x, y) = |\Gamma(x) \cap \Gamma(y)|$, where $\Gamma(x)$ is the set of x's neighbors in the homogeneous co-authorship network.
- Jaccard: $score(x, y) = \frac{|\Gamma(x)\cap\Gamma(y)|}{|\Gamma(x)\cup\Gamma(y)|}$.

- Adamic: $score(x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{log|\Gamma(z)|}$.
- Katz: $score(x, y) = \sum_{l=1 \ldots \infty} \beta^l \cdot |path_{x,y}^{\langle l \rangle}|$, where β is the damping factor to penalize longer paths and $path_{x,y}^{\langle l \rangle}$ is the set of length-l paths between x and y. We consider paths up to length 3 here.

Attribute-based similarity is also tested here by calculating the cosine similarity (Cosine) of the attribute vectors corresponding to the different authors, the attribute vector v for an author a is a vector in k-space where k is the number of unique attributes in our dataset and $v_i = 1$ if a is linked to the i^{th} attribute, $v_i = 0$ otherwise.

We also test our method against the meta path-based PathPredict framework described in [14]. In the tests conducted, we use the hybrid topological features defined in the previous work and learn the weights assigned to these features separately for two groups of authors: (1) highly productive authors (authors who have published more than 10 papers within $[1991, 2007]$), and (2) less productive authors (authors with less than 10 papers in $[1991, 2007]$).

Finally, we test different variants of our proposed algorithm. RW_GL is used to denote our random walk algorithm based on the new graph with edges weighted according to global and local importance measures alone, trivial self-citation links are not included in the modified graph. RW_FR runs the proposed algorithm on the same graph as RW_GL with edges weighted according to frequency and recency importance measures only. RW_GLFR denotes the same algorithm run on the new graph with edges weighted using all importance measures, again trivial self-citation links are not added to the graph. RW_All is the algorithm based on the new graph with edges weighted using all importance measures and with trivial self-citation links included. We also run a random walk with restart on the original heterogeneous bibliographic network with links from each node weighted uniformly, we refer to this as RW_Naive.

C. Comparison of Different Methods

Table 1 lists the scores of the different methods based on precision and recall. Among the different structure-based measures, Common Neighbors and Jaccard's Coefficient performed best. The Katz index was the least effective among this group. Overall, the attribute-based Cosine Similarity method performed most poorly. Unsurprisingly, the straight-forward implementation of random walk with restart on the original network did not produce very good results either.

As expected, the meta path-based PathPredict framework gave results that were better than all the other methods based on network topology and attributes. It even achieved a better recall score than RW_GL although its precision score could not beat that of any of the variants of our proposed method. This seems to indicate that the important relations defined in [14] can be better highlighted when captured in a modified graph with edges biased according to measures of importance.

Running a random walk on the new graph weighted using global and local importance measures alone already provides results that are better than most of the other methods. In our dataset, however, it is interesting to observe that using just frequency and recency measure to weight the links is already sufficient and RW_FR actually scores slightly better than RW_GLFR. Finally, adding trivial self-citation links to the network helps improve the predictive accuracy further.

	Prec@5	**Rec@5**	**Prec@10**	**Rec@10**
ComNeigh	0.059	0.117	0.044	0.166
Jaccard	0.060	0.120	0.043	0.163
Adamic	0.058	0.111	0.042	0.151
Katz	0.052	0.101	0.039	0.141
Cosine	0.030	0.054	0.023	0.081
PathPredict	**0.064**	**0.145**	**0.049**	**0.205**
RW_Naive	0.043	0.093	0.035	0.145
RW_GL	**0.070**	**0.137**	**0.052**	**0.189**
RW_FR	**0.073**	**0.147**	**0.056**	**0.208**
RW_GLFR	**0.072**	**0.147**	**0.055**	**0.206**
RW_All	**0.080**	**0.161**	**0.058**	**0.221**

TABLE I
COMPARISON OF RESULTS BETWEEN THE DIFFERENT METHODS. RESULTS OF THE VARIANTS OF OUR PROPOSED ALGORITHM AND THE TOP BENCHMARK ALGORITHM ARE IN BOLD-FACE.

Notice that in general, all methods scored relatively low, this is due to the fact that the test dataset is comprised of the 2,505 authors who have published papers in the WWW conferences from years 2001 to 2008 only. All other authors that should have been linked to these authors are not included, thus the dataset loses a lot of structural information.

D. Acceptable Parameter Values

Our proposed algorithm takes several parameters as input, namely: λ, α, β, γ, δ, and c. We discuss here the values that were found to be acceptable for the different parameters in the studied dataset and their effect on the prediction outcome.

The optimal set of values for the different parameters may vary across data sets. In our experiments, we tested different combinations of values for the parameters and we discuss here the most acceptable values in terms of predictive accuracy.

Effect of λ setting. λ controls the overall importance of structural links over attribute links. If λ is large, the random walk is propagated primarily through the attribute links. On the other hand, if λ is small, then structural links play a more important role in the random walker's traversal of the graph. The most acceptable value for λ in our experiments is 0.6. This is quite intuitive since attribute links generally outnumber structural links and a larger value for λ distributes the weights more evenly across the two kinds of links. However, we find that the ideal value for λ in our experiments is the same as the one reported by [19] in their paper even though our graph contains more attribute nodes. This seems to indicate that only important attribute nodes should be prioritized.

Effect of setting the different importance parameters. For the parameters that determined the trade-off among the different importance measures, the best setting found was $\alpha = 0.20$, $\beta = 0.10$, $\gamma = 0.35$, and $\delta = 0.35$. It is interesting to note that the importance measures based on

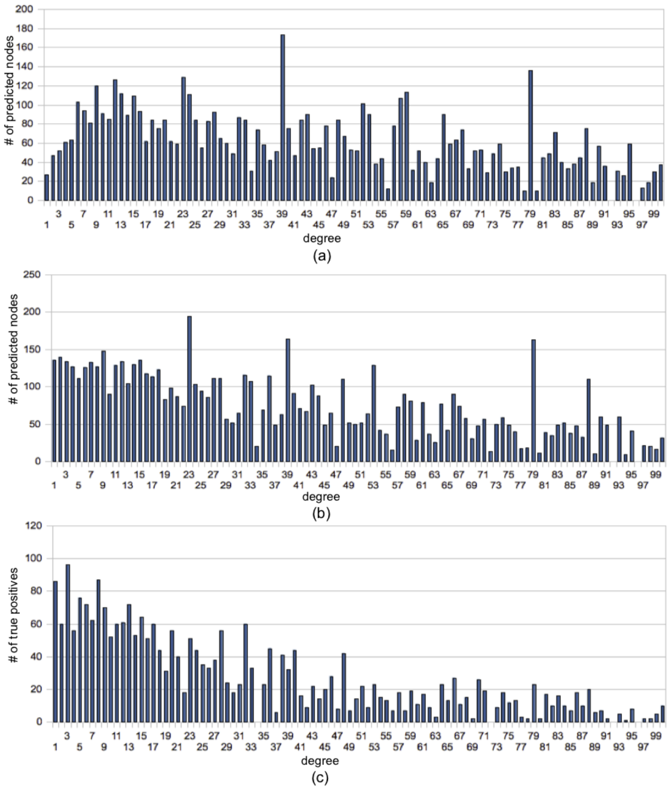

Fig. 4. The graph in (a) shows the degree distribution of the top-10 predicted co-authors for all authors using the RW_Naive algorithm, graph (b) shows the same information for the RW_All algorithm while graph (c) shows the degree distribution of real future co-authors of all authors in the dataset.

frequency and recency of interaction play a more crucial role in defining the overall importance of a link. It is also worth noticing that while multiple interactions are important, the recency of interaction is equally important in our setting. Finally, we observe that more emphasis is placed on global importance when determining the importance of a link. All four importance measures contribute to the identification of important links, albeit at varying degrees.

Effect of c setting. We find that the best value for c is 0.7. This means that the local neighborhood is important when searching for potential co-author candidates. We notice, however, that the ideal value for c in our experiments is less than the ones reported by [19] in their link recommendation experiments. This may be due to the fact that the richer information encoded in our new graph helps discriminate real potential co-authors from the other nodes and thus the random walk can be extended to a slightly more global neighborhood.

E. Degree Distribution of Candidate Co-authors

Fig. 4 shows the degree distribution of the predicted co-authors and the real distribution of actual co-authors. Graph (a) shows the degree distribution of the co-authors predicted by RW_Naive algorithm on the original graph. One can see that the random walk tends to miss nodes with very low degrees, which is its natural behavior [4]. The second graph shows the degree distribution of the nodes predicted as potential co-authors by the RW_All algorithm, here there is less bias towards nodes with low degrees which is a property we wish to find in a good algorithm since in the real world authors may co-author with both high degree and low degree authors. Finally, graph (c) shows the degree distribution of real future co-authors; a qualitative assessment shows that graph (b) and (c) are more similar.

F. Case Study

To contrast and compare the different methods further, we study the top-10 recommended co-authors for Juanzi Li, who is a well-known researcher from Tsinghua University. Table 2 displays the list of recommended co-authors for the different methods. We find that the structure-based methods, namely Common Neighbors, Jaccard's Coefficient, and Adamic-Adar Score, produce lists that are similar to one another. For this particular researcher, our method's predicted list of co-authors is also quite similar to the ones produced by the structure-based methods. However, the true co-authors are ranked higher in our list which suggests that attribute information and importance measures can help in further discriminating the true co-authors from the false candidates – from a set of authors that have strong structural links to the query author.

The method based on the Katz Index also made the same number of correct prediction as our proposed method, this is to be expected in some cases as the Katz Index is based on the number of paths between two nodes and this is analogous to the random walk probability from one node to another since the more paths there are between two nodes the higher the random walk probability from one to the other. However, from the overall score of the predictive accuracy of the Katz-based method, we know that considering the number of paths between two nodes does not always produce the best results.

In our case study, the attribute-based method only made three correct predictions, this is indicative of the fact that although attribute similarity can help in identifying future collaboration, structural information is still important. For the author Juanzi Li, four of the top ten predicted co-authors by the algorithm based on the PathPredict framework were correct. This is one better than the predictions made by the method based on cosine similarity of attributes but slightly worse than the predictions of the structure based methods. This highlights a problem with the PathPredict approach. Although it is able to learn, in general, the weights or importance of the different meta paths, there are cases wherein a query author is connected to many false as well as true candidates by important meta paths and in such cases it can be difficult for the algorithm to distinguish the true co-authors from the false ones (*e.g.* authors that are distant from the query author in the social graph but who share many attributes with the author or popular authors).

G. Average Run-time of Implementation

We computed the average number of iterations required for the random walk to converge on all the author nodes in our dataset with restart probability $c \in \{0.1, 0.2, ..., 0.9\}$.

As expected, a smaller value for c results in more iterations

RW_All (6)	*ComNeigh (5)*	*Jaccard (5)*	*Adamic (5)*	*Katz (6)*	*Cosine (3)*	*PathPredict (4)*
Duo Zhang	**Duo Zhang**	**Duo Zhang**	**Duo Zhang**	**Duo Zhang**	**Duo Zhang**	**Duo Zhang**
Kuo Zhang	Kuo Zhang	Kuo Zhang	Kuo Zhang	**MingCai Hong**	Risto Gligorov	Kuo Zhang
Jing Zhang	Min Zhang	**Limin Yao**	**Limin Yao**	**Jing Zhang**	Zharko Aleksovski	**MingCai Hong**
MingCai Hong	**Limin Yao**	**MingCai Hong**	Hwee Tou Ng	**Limin Yao**	**Jing Zhang**	**Limin Yao**
Limin Yao	Hwee Tou Ng	Ming Zhang	Wei Wei	Kuo Zhang	Yunxiao Ma	**Jing Zhang**
Qiong Luo	Wei Wei	Hwee Tou Ng	**MingCai Hong**	**Qiong Luo**	**Limin Yao**	Hang Li
Min Zhang	**MingCai Hong**	Wei Wei	**Qiong Luo**	Hang Li	Warner ten Kate	Hwee Tou Ng
Yunhao Liu	**Qiong Luo**	**Jing Zhang**	**Jing Zhang**	**Yunhao Liu**	Ming Mao	Min Zhang
Wei Wei	**Jing Zhang**	Yunbo Cao	Yunbo Cao	Wei Wei	Krisztian Balog	Wei Wei
Hang Li	Shenghuo Zhu	**Yunhao Liu**	Hang Li	Wei-Ying Ma	Mikhail Bilenko	Yunbo Cao

TABLE II

TOP TEN PREDICTED CO-AUTHORS OF DIFFERENT METHODS FOR JUANZI LI. TRUE CO-AUTHORS ARE DISPLAYED IN BOLD-FACE, AND NUMBER OF CORRECT PREDICTIONS ARE INSERTED IN PARENTHESES.

since the random walk explores a more global neighborhood. When the restart probability is 0.1, the algorithm takes a little more than 25 iterations while it only requires around 5 iterations when $c = 0.9$. It is worth noticing that in our network, with close to $140k$ nodes, the random walk with $c = 0.1$ takes only 25 iterations on average to run. On our test machine, with a 2.26 Ghz dual core processor and 2 Gb of memory, it took an average of 1.12 seconds to run the Java implementation of the algorithm with restart probability $c = 0.1$ for all authors. From this result, we can intuit that the proposed method should run reasonably well for graphs comprised of a few hundred thousand nodes to tens of millions of nodes. For very large graphs, one can choose to use the algorithm described in [18] to speed up the random walk.

VI. CONCLUSION AND FUTURE WORK

In this work, we defined an algorithm based on random walks for link prediction on a heterogeneous bibliographic information network. Specifically, we have shown the algorithm's efficiency can be increased by running it on a modified heterogeneous bibliographic network. We have also shown that global and local importance measures, as well as the frequency and recency of interaction across links are all useful, within varying degrees, for identifying the importance of an edge.

In future work, we wish to consider various loss [3] and growth functions when calculating the frequency and recency importance measures. The effectiveness of the various functions can be tested on different bibliographic networks for us to gain a better understanding of the decay and growth of relationships in real world bibliographic networks.

Although multiple parameters allow a higher level of fine-tuning, identifying the proper weight assignment can be a problem. We wish to study ways to approximate ideal values for parameters from certain network characteristics.

The authors would also like to test the performance of the algorithm on the entire DBLP network. A generalized version of the algorithm that works well on networks from various domains should also be considered.

The proposed model can also be turned into a probabilistic model to capture the correlations among links or nodes of the graph. Furthermore, the model may be extended to tackle similar problems such as the problem where the time of future link creation is predicted; a recent work is found in [15].

REFERENCES

[1] L. A. Adamic and E. Adar. Friends and neighbors on the web. *Social Networks*, 25:211–230, 2001.

[2] A. S. Aytuna, A. Gursoy, and O. Keskin. Prediction of protein–protein interactions by combining structure and sequence conservation in protein interfaces. *Bioinformatics*, 21(12):2850–2855, April 2005.

[3] L. Backstrom and J. Leskovec. Supervised random walks: Predicting and recommending links in social networks. In *Proc. of WSDM '11*, pages 635–644, 2011.

[4] P. Blanchard and D. Volchenkov. *Random Walks and Diffusions on Graphs and Databases: An Introduction*. Springer, 2011.

[5] G. Cabunducan, R. Castillo, and J. B. Lee. Voting behavior analysis in the election of wikipedia admins. In *Proc. of ASONAM '11*, 2011.

[6] L. Getoor and C. P. Diehl. Link mining: a survey. *SIGKDD Explor. Newsletter*, 7:3–12, 2005.

[7] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *Proc. of SDM '06*, 2006.

[8] M. Hue, M. Riffle, J. P. Vert, and W. S. Noble. Large-scale prediction of protein-protein interactions from structures. *BMC Bioinformatics*, 11:144, March 2010.

[9] H. Kashima and N. Abe. A parameterized probabilistic model of network evolution for supervised link prediction. In *Proc. of ICDM '06*, 2006.

[10] V. Leroy, B. B. Cambazoglu, and F. Bonchi. Cold start link prediction. In *Proc. of KDD '10*, pages 393–402, 2010.

[11] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proc. of CIKM '03*, pages 556–559, 2003.

[12] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. Mining sequential patterns by pattern-growth: The prefixspan approach. *TKDE*, 16(11):1424–1440, 2004.

[13] E. Spyropoulou and T. De Bie. Interesting multi-relational patterns. In *Proc. of ICDM '11*, pages 675–684, 2011.

[14] Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han. Co-author relationship prediction in heterogeneous bibliographic networks. In *Proc. of ASONAM '11*, pages 121–128, 2011.

[15] Y. Sun, J. Han, C. C. Aggarwal, and N. V. Chawla. When will it happen? - relationship prediction in heterogeneous information networks. In *Proc. of WSDM '12*, 2012.

[16] Y. Sun, Y. Yu, and J. Han. Kdd '09 slide presentation of the paper "ranking-based clustering of heterogeneous information networks with star network schema", 2009.

[17] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. Arnetminer: extraction and mining of academic social networks. In *Proc. of KDD '08*, pages 990–998, 2008.

[18] H. Tong, C. Faloutsos, and J.-Y. Pan. Fast random walk with restart and its applications. In *Proc. of ICDM '06*, pages 613–622, 2006.

[19] Z. Yin, M. Gupta, T. Weninger, and J. Han. A unified framework for link recommendation using random walks. In *Proc. of ASONAM '10*, pages 152–159, 2010.