The Link Prediction Problem for Social Networks*

David Liben-Nowell†
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139 USA
dln@theory.lcs.mit.edu

Jon Kleinberg‡
Department of Computer Science
Cornell University
Ithaca, NY 14853 USA
kleinber@cs.cornell.edu

January 8, 2004

Abstract

Given a snapshot of a social network, can we infer which new interactions among its members are likely to occur in the near future? We formalize this question as the *link prediction problem*, and develop approaches to link prediction based on measures for analyzing the "proximity" of nodes in a network. Experiments on large co-authorship networks suggest that information about future interactions can be extracted from network topology alone, and that fairly subtle measures for detecting node proximity can outperform more direct measures.

1 Introduction

As part of the recent surge of research on large, complex networks and their properties, a considerable amount of attention has been devoted to the computational analysis of *social networks*—structures whose nodes represent people or other entities embedded in a social context, and whose edges represent interaction, collaboration, or influence between entities. Natural examples of social networks include the set of all scientists in a particular discipline, with edges joining pairs who have co-authored papers; the set of all employees in a large company, with edges joining pairs working on a common project; or a collection of business leaders, with edges joining pairs who have served together on a corporate board of directors. The availability of large, detailed datasets encoding such networks has stimulated extensive study of their basic properties, and the identification of recurring structural features. (See, for example, the work of Watts and Strogatz [28], Watts [27], Grossman [9], Newman [19], and Adamic and Adar [1], or, for a thorough recent survey, Newman [20].)

Social networks are highly dynamic objects; they grow and change quickly over time through the addition of new edges, signifying the appearance of new interactions in the underlying social structure. Understanding the mechanisms by which they evolve is a fundamental question that is still not well understood, and it forms the motivation for our work here. We define and study a basic computational problem underlying social network evolution, the *link prediction problem*:

*An abbreviated version of this paper appears in *Proceedings of the Twelfth Annual ACM International Conference on Information and Knowledge Management (CIKM'03)*, November 2003, pp. 556–559.

†Supported in part by an NSF Graduate Research Fellowship.

‡Supported in part by a David and Lucile Packard Foundation Fellowship and NSF ITR Grant IIS-0081334.

Given a snapshot of a social network at time t, we seek to accurately predict the edges that will be added to the network during the interval from time t to a given future time t'.

In effect, the link prediction problem asks: to what extent can the evolution of a social network be modeled using features *intrinsic to the network itself?* Consider a co-authorship network among scientists, for example. There are many reasons, exogenous to the network, why two scientists who have never written a paper together will do so in the next few years: for example, they may happen to become geographically close when one of them changes institutions. Such collaborations can be hard to predict. But one also senses that a large number of new collaborations are hinted at by the topology of the network: two scientists who are "close" in the network will have colleagues in common, and will travel in similar circles; this suggests that they themselves are more likely to collaborate in the near future. Our goal is to make this intuitive notion precise, and to understand which measures of "proximity" in a network lead to the most accurate link predictions. We find that a number of proximity measures lead to predictions that outperform chance by factors of 40 to 50, indicating that the network topology does indeed contain latent information from which to infer future interactions. Moreover, certain fairly subtle measures—involving infinite sums over paths in the network—often outperform more direct measures, such as shortest-path distances and numbers of shared neighbors.

We believe that a primary contribution of the present paper is in the area of network evolution models. While there has been a proliferation of such models in recent years—see, for example, the work of Jin et al. [11], Barabasi et al. [2], and Davidsen et al. [5] for recent work on collaboration networks, or the survey of Newman [20]—they have generally been evaluated only by asking whether they reproduce certain global structural features observed in real networks. As a result, it has been difficult to evaluate and compare different approaches on a principled footing. Link prediction, on the other hand, offers a very natural basis for such evaluations: *a network model is useful to the extent that it can support meaningful inferences from observed network data.* One sees a related approach in recent work of Newman [17], who considers the correlation between certain network growth models and data on the appearance of edges of co-authorship networks.

In addition to its role as a basic question in social network evolution, the link prediction problem could be relevant to a number of interesting current applications of social networks. Increasingly, for example, researchers in artificial intelligence and data mining have argued that a large organization, such as a company, can benefit from the interactions within the informal social network among its members; these serve to supplement the official hierarchy imposed by the organization itself [13, 23]. Effective methods for link prediction could be used to analyze such a social network, and suggest promising interactions or collaborations that have not yet been utilized within the organization. In a different vein, research in security has recently begun to emphasize the role of social network analysis, largely motivated by the problem of monitoring terrorist networks; link prediction in this context allows one to conjecture that particular individuals are working together even though their interaction has not been directly observed [14].

The link prediction problem is also related to the problem of inferring missing links from an observed network: in a number of domains, one constructs a network of interactions based on observable data and then tries to infer additional links that, while not directly visible, are likely to exist [8, 22, 26]. This line of work differs from our problem formulation in that it works with a static snapshot of a network, rather than considering network evolution; it also tends to take into account specific attributes of the nodes in the network, rather than evaluating the power of prediction methods based purely on the graph structure.

We now turn to a description of our experimental setup, in Section 2. Our primary focus is on

	training period			Core		
	authors	papers	edges	authors	$\lvert E_{old}\rvert$	$\lvert E_{new}\rvert$
`astro-ph`	5343	5816	41852	1561	6178	5751
`cond-mat`	5469	6700	19881	1253	1899	1150
`gr-qc`	2122	3287	5724	486	519	400
`hep-ph`	5414	10254	17806	1790	6654	3294
`hep-th`	5241	9498	15842	1438	2311	1576

Figure 1: The five sections of the arXiv from which co-authorship networks were constructed: `astro-ph` (astrophysics), `cond-mat` (condensed matter), `gr-qc` (general relativity and quantum cosmology), `hep-ph` (high energy physics—phenomenology), and `hep-th` (high energy physics—theory). The set Core is the subset of the authors who have written at least $\kappa_{training} = 3$ papers during the training period and $\kappa_{test} = 3$ papers during the test period. The sets E_{old} and E_{new} denote edges between Core authors which first appear during the training and test periods, respectively.

understanding the relative effectiveness of network proximity measures adapted from techniques in graph theory, computer science, and the social sciences, and we review a large number of such techniques in Section 3. Finally, we discuss the results of our experiments in Section 4.

2 Data and Experimental Setup

Suppose we have a social network $G = \langle V, E\rangle$ in which each edge $e = \langle u, v\rangle \in E$ represents an interaction between u and v that took place at a particular time $t(e)$. We record multiple interactions between u and v as parallel edges, with potentially different time-stamps. For two times $t < t'$, let $G[t, t']$ denote the subgraph of G consisting of all edges with a time-stamp between t and t'. Here, then, is a concrete formulation of the link prediction problem. We choose four times $t_0 < t'_0 < t_1 < t'_1$, and give an algorithm access to the network $G[t_0, t'_0]$; it must then output a list of edges, not present in $G[t_0, t'_0]$, that are predicted to appear in the network $G[t_1, t'_1]$. We refer to $[t_0, t'_0]$ as the *training interval* and $[t_1, t'_1]$ as the *test interval*.

Of course, social networks grow through the addition of nodes as well as edges, and it is not sensible to seek predictions for edges whose endpoints are not present in the training interval. Thus, in evaluating link prediction methods, we will generally use two parameters $\kappa_{training}$ and κ_{test} (each set to 3 below), and define the set Core to be all nodes incident to at least $\kappa_{training}$ edges in $G[t_0, t'_0]$ and at least κ_{test} edges in $G[t_1, t'_1]$. We will then evaluate how accurately the new edges between elements of Core can be predicted.

We now describe our experimental setup more specifically. We work with five co-authorship networks G, obtained from the author lists of papers at five sections of the physics e-Print arXiv, `www.arxiv.org`. (See Figure 1 for statistics on the size of each of these five networks.) Some heuristics were used to deal with occasional syntactic anomalies; and authors were identified by first initial and last name, a process that introduces a small amount of noise due to multiple authors with the same identifier [18]. The errors introduced by this process appear to be minor.

Now consider any one of these five graphs. We define the training interval to be the three years $[1994, 1996]$, and the test interval to be $[1997, 1999]$. We denote the subgraph $G[1994, 1996]$ on the training interval by $G_{collab} := \langle A, E_{old}\rangle$, and use E_{new} to denote the set of edges $\langle u, v\rangle$ such that $u, v \in A$, and u, v co-author a paper during the test interval but not the training interval—these are the new interactions we are seeking to predict.

graph distance	(negated) length of shortest path between x and y
common neighbors	$\lvert\Gamma(x) \cap \Gamma(y)\rvert$
Jaccard's coefficient	$\frac{\lvert\Gamma(x)\cap\Gamma(y)\rvert}{\lvert\Gamma(x)\cup\Gamma(y)\rvert}$
Adamic/Adar	$\sum_{z\in\Gamma(x)\cap\Gamma(y)} \frac{1}{\log\lvert\Gamma(z)\rvert}$
preferential attachment	$\lvert\Gamma(x)\rvert \cdot \lvert\Gamma(y)\rvert$
Katz$_\beta$	$\sum_{\ell=1}^{\infty} \beta^\ell \cdot \lvert\mathsf{paths}_{x,y}^{\langle\ell\rangle}\rvert$ where $\mathsf{paths}_{x,y}^{\langle\ell\rangle}$:= {paths of length exactly ℓ from x to y} weighted: $\mathsf{paths}_{x,y}^{\langle 1\rangle}$:= number of collaborations between x, y. unweighted: $\mathsf{paths}_{x,y}^{\langle 1\rangle} := 1$ iff x and y collaborate.
hitting time	$-H_{x,y}$
stationary-normed	$-H_{x,y} \cdot \pi_y$
commute time	$-(H_{x,y} + H_{y,x})$
stationary-normed	$-(H_{x,y} \cdot \pi_y + H_{y,x} \cdot \pi_x)$ where $H_{x,y}$:= expected time for random walk from x to reach y π_y := stationary distribution weight of y (proportion of time the random walk is at node y)
rooted PageRank$_\alpha$	stationary distribution weight of y under the following random walk: with probability α, jump to x. with probability $1-\alpha$, go to random neighbor of current node.
SimRank$_\gamma$	$\begin{cases} 1 & \text{if } x = y \\ \gamma \cdot \frac{\sum_{a\in\Gamma(x)}\sum_{b\in\Gamma(y)} \mathsf{score}(a,b)}{\lvert\Gamma(x)\rvert\cdot\lvert\Gamma(y)\rvert} & \text{otherwise} \end{cases}$

Figure 2: Values for $\mathsf{score}(x, y)$ under various predictors; each predicts pairs $\langle x, y\rangle$ in descending order of $\mathsf{score}(x, y)$. The set $\Gamma(x)$ consists of the neighbors of the node x in G_{collab}.

Evaluating a link predictor. Each link predictor p that we consider outputs a ranked list L_p of pairs in $A \times A - E_{old}$; these are predicted new collaborations, in decreasing order of confidence. For our evaluation, we focus on the set Core, so we define $E^*_{new} := E_{new} \cap (\mathsf{Core} \times \mathsf{Core})$ and $n := |E^*_{new}|$. Our performance measure for predictor p is then determined as follows: from the ranked list L_p, we take the first n pairs in $\mathsf{Core} \times \mathsf{Core}$, and determine the size of the intersection of this set of pairs with the set E^*_{new}.

3 Methods for Link Prediction

In this section, we survey an array of methods for link prediction. All the methods assign a connection weight $\mathsf{score}(x, y)$ to pairs of nodes $\langle x, y\rangle$, based on the input graph G_{collab}, and then produce a ranked list in decreasing order of $\mathsf{score}(x, y)$. Thus, they can be viewed as computing a measure of proximity or "similarity" between nodes x and y, relative to the network topology. In general, the methods are adapted from techniques used in graph theory and social network analysis; in a number of cases, these techniques were not designed to measure node-to-node similarity, and hence need to be modified for this purpose. Figure 2 summarizes most of these measures; below we

discuss them in more detail. We note that some of these measures are designed only for connected graphs; since each graph G_{collab} that we consider has a *giant component*—a single component containing most of the nodes—it is natural to restrict the predictions for these measures to this component.

Perhaps the most basic approach is to rank pairs $\langle x, y\rangle$ by the length of their shortest path in G_{collab}. Such a measure follows the notion that collaboration networks are "small worlds," in which individuals are related through short chains [18]. (In keeping with the notion that we rank pairs in *decreasing* order of $\mathsf{score}(x, y)$, we define $\mathsf{score}(x, y)$ here to be the negative of the shortest path length.) Pairs with shortest-path distance equal to 1 are joined by an edge in G_{collab}, and hence they belong to the training edge set E_{old}. For all of our graphs G_{collab}, there are well more than n pairs at shortest-path distance two, so our shortest-path predictor simply selects a random subset of these distance-two pairs.

Methods based on node neighborhoods. For a node x, let $\Gamma(x)$ denote the set of neighbors of x in G_{collab}. A number of approaches are based on the idea that two nodes x and y are more likely to form a link in the future if their sets of neighbors $\Gamma(x)$ and $\Gamma(y)$ have large overlap; this follows the natural intuition that such nodes x and y represent authors with many colleagues in common, and hence are more likely to come into contact themselves. Jin et al. [11] and Davidsen et al. [5] have defined abstract models for network growth using this principle, in which an edge $\langle x, y\rangle$ is more likely to form if edges $\langle x, z\rangle$ and $\langle z, y\rangle$ are already present for some z.

• *Common neighbors.* The most direct implementation of this idea for link prediction is to define $\mathsf{score}(x, y) := |\Gamma(x) \cap \Gamma(y)|$, the number of neighbors that x and y have in common. Newman [17] has computed this quantity in the context of collaboration networks, verifying a correlation between the number of common neighbors of x and y at time t, and the probability that they will collaborate in the future.

• *Jaccard's coefficient and Adamic/Adar.* The Jaccard coefficient—a commonly used similarity metric in information retrieval [24]—measures the probability that both x and y have a feature f, for a randomly selected feature f that *either* x or y has. If we take "features" here to be neighbors in G_{collab}, this leads to the measure $\mathsf{score}(x, y) := |\Gamma(x) \cap \Gamma(y)|/|\Gamma(x) \cup \Gamma(y)|$. Adamic and Adar [1] consider a related measure, in the context of deciding when two personal home pages are strongly "related." To do this, they compute features of the pages, and define the similarity between two pages to be

$$\sum_{z \,:\, \text{feature shared by } x,\, y} \frac{1}{\log(\mathsf{frequency}(z))}.$$

This refines the simple counting of common features by weighting rarer features more heavily. This suggests the measure $\mathsf{score}(x, y) := \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log |\Gamma(z)|}$.

• *Preferential attachment* has received considerable attention as a model of the growth of networks [16]. The basic premise is that the probability that a new edge involves node x is proportional to $|\Gamma(x)|$, the current number of neighbors of x. Newman [17] and Barabasi et al. [2] have further proposed, on the basis of empirical evidence, that the probability of co-authorship of x and y is correlated with the product of the number of collaborators of x and y. This corresponds to the measure $\mathsf{score}(x, y) := |\Gamma(x)| \cdot |\Gamma(y)|$.

Methods based on the ensemble of all paths. A number of methods refine the notion of shortest-path distance by implicitly considering the ensemble of *all* paths between two nodes.

• *Katz* [12] defines a measure that directly sums over this collection of paths, exponentially damped by length to count short paths more heavily. This leads to the measure

$$\mathsf{score}(x, y) := \sum_{\ell=1}^{\infty} \beta^{\ell} \cdot |\mathsf{paths}_{x,y}^{\langle \ell \rangle}|,$$

where $\mathsf{paths}_{x,y}^{\langle \ell \rangle}$ is the set of all length-ℓ paths from x to y. (A very small β yields predictions much like common neighbors, since paths of length three or more contribute very little to the summation.) One can verify that the matrix of scores is given by $(I - \beta M)^{-1} - I$, where M is the adjacency matrix of the graph. We consider two variants of this Katz measure: (1) *unweighted*, in which $\mathsf{paths}_{x,y}^{\langle 1 \rangle} = 1$ if x and y have collaborated and 0 otherwise, and (2) *weighted*, in which $\mathsf{paths}_{x,y}^{\langle 1 \rangle}$ is the number of times that x and y have collaborated.

• *Hitting time, PageRank, and variants.* A *random walk* on G_{collab} starts at a node x, and iteratively moves to a neighbor of x chosen uniformly at random. The *hitting time* $H_{x,y}$ from x to y is the expected number of steps required for a random walk starting at x to reach y. Since the hitting time is not in general symmetric, it is also natural to consider the *commute time* $C_{x,y} := H_{x,y} + H_{y,x}$. Both of these measures serve as natural proximity measures, and hence (negated) can be used as $\mathsf{score}(x, y)$.

One difficulty with hitting time as a measure of proximity is that $H_{x,y}$ is quite small whenever y is a node with a large *stationary probability* π_y, regardless of the identity of x. To counterbalance this phenomenon, we also consider *normalized* versions of the hitting and commute times, by defining $\mathsf{score}(x, y) := -H_{x,y} \cdot \pi_y$ or $\mathsf{score}(x, y) := -(H_{x,y} \cdot \pi_y + H_{y,x} \cdot \pi_x)$.

Another difficulty with these measures is their sensitive dependence to parts of the graph far away from x and y, even when x and y are connected by very short paths. A way of counteracting this is to allow the random walk from x to y to periodically "reset," returning to x with a fixed probability α at each step; in this way, distant parts of the graph will almost never be explored. Random resets form the basis of the *PageRank* measure for Web pages [3], and we can adapt it for link prediction as follows: Define $\mathsf{score}(x, y)$ under the *rooted PageRank* measure to be the stationary probability of y in a random walk that returns to x with probability α each step, moving to a random neighbor with probability $1 - \alpha$.

• *SimRank* [10] is a fixed point of the following recursive definition: two nodes are similar to the extent that they are joined to similar neighbors. Numerically, this is specified by defining $\mathsf{similarity}(x, x) := 1$ and

$$\mathsf{similarity}(x, y) := \gamma \cdot \frac{\sum_{a \in \Gamma(x)} \sum_{b \in \Gamma(y)} \mathsf{similarity}(a, b)}{|\Gamma(x)| \cdot |\Gamma(y)|}$$

for some $\gamma \in [0, 1]$. We then define $\mathsf{score}(x, y) := \mathsf{similarity}(x, y)$. SimRank can also be interpreted in terms of a random walk on the collaboration graph: it is the expected value of γ^{ℓ}, where ℓ is a random variable giving the time at which random walks started from x and y first meet.

Higher-level approaches. We now discuss three "meta-approaches" that can be used in conjunction with any of the methods discussed above.

• *Low-rank approximation.* Since the adjacency matrix M can be used to represent the graph G_{collab}, all of our link prediction methods have an equivalent formulation in terms of this matrix M. In some cases, this was noted explicitly above (for example in the case of the Katz similarity score); but in many cases the matrix formulation is quite natural. For example, the common neighbors

method consists simply of mapping each node x to its row $r(x)$ in M, and then defining $\mathsf{score}(x, y)$ to be the inner product of the rows $r(x)$ and $r(y)$.

A common general technique when analyzing the structure of a large matrix M is to choose a relatively small number k and compute the rank-k matrix M_k that best approximates M with respect to any of a number of standard matrix norms. This can be done efficiently using the singular value decomposition, and it forms the core of methods like *latent semantic analysis* in information retrieval [6]. Intuitively, working with M_k rather than M can be viewed as a type of "noise-reduction" technique that generates most of the structure in the matrix but with a greatly simplified representation.

In our experiments, we investigate three applications of low-rank approximation: (i) ranking by the Katz measure, in which we use M_k rather than M in the underlying formula; (ii) ranking by common neighbors, in which we score by inner products of rows in M_k rather than M; and—most simply of all—(iii) defining $\mathsf{score}(x, y)$ to be the $\langle x, y \rangle$ entry in the matrix M_k.

• *Unseen bigrams.* Link prediction is akin to the problem of estimating frequencies for *unseen bigrams* in language modeling—pairs of words that co-occur in a test corpus, but not in the corresponding training corpus (see, e.g., the work of Essen and Steinbiss [7]). Following ideas proposed in that literature [15, for example], we can augment our estimates for $\mathsf{score}(x, y)$ using values of $\mathsf{score}(z, y)$ for nodes z that are "similar" to x. Specifically, we adapt this to the link prediction problem as follows. Suppose we have values $\mathsf{score}(x, y)$ computed under one of the measures above. Let $S_x^{\langle\delta\rangle}$ denote the δ nodes most related to x under $\mathsf{score}(x, \cdot)$, for a parameter $\delta \in \mathbb{Z}^+$. We then define enhanced scores in terms of the nodes in this set:

$$\begin{aligned} \mathsf{score}^*_{unweighted}(x, y) &:= \left|\{z : z \in \Gamma(y) \cap S_x^{\langle\delta\rangle}\}\right| \\ \mathsf{score}^*_{weighted}(x, y) &:= \sum_{z \in \Gamma(y) \cap S_x^{\langle\delta\rangle}} \mathsf{score}(x, z). \end{aligned}$$

• *Clustering.* One might seek to improve on the quality of a predictor by deleting the more "tenuous" edges in G_{collab} through a clustering procedure, and then running the predictor on the resulting "cleaned-up" subgraph. Consider a measure computing values for $\mathsf{score}(x, y)$. We compute $\mathsf{score}(u, v)$ for all edges in E_{old}, and delete the $(1 - \rho)$ fraction of these edges for which the score is lowest. We now recompute $\mathsf{score}(x, y)$ for all pairs $\langle x, y \rangle$ on this subgraph; in this way we determine node proximities using only edges for which the proximity measure itself has the most confidence.

4 Results and Discussion

As discussed in Section 1, many collaborations form (or fail to form) for reasons outside the scope of the network; thus the raw performance of our predictors is relatively low. To more meaningfully represent predictor quality, we use as our baseline a *random predictor* which simply randomly selects pairs of authors who did not collaborate in the training interval. A random prediction is correct with probability between 0.15% (`cond-mat`) and 0.48% (`astro-ph`). Figures 3 and 4 show each predictor's performance on each arXiv section, in terms of the factor improvement over random. Figures 5, 6, and 7 show the average relative performance of several different predictors versus three baseline predictors—the random predictor, the graph distance predictor, and the common neighbors predictor. There is no single clear winner among the techniques, but we see that a number of methods significantly outperform the random predictor, suggesting that there is indeed useful information contained in the network topology alone. The Katz measure and its variants based on clustering and low-rank approximation perform consistently well; on three of the five

predictor		astro-ph	cond-mat	gr-qc	hep-ph	hep-th
probability that a random prediction is correct		0.475%	0.147%	0.341%	0.207%	0.153%
graph distance (all distance-two pairs)		*9.6*	*25.3*	*21.4*	*12.2*	*29.2*
common neighbors		**18.0**	**41.1**	**27.2**	**27.0**	**47.2**
preferential attachment		4.7	6.1	7.6	*15.2*	7.5
Adamic/Adar		*16.8*	**54.8**	**30.1**	**33.3**	**50.5**
Jaccard		*16.4*	**42.3**	19.9	**27.7**	*41.7*
SimRank	$\gamma = 0.8$	*14.6*	*39.3*	*22.8*	*26.1*	*41.7*
hitting time		6.5	23.8	*25.0*	3.8	13.4
hitting time—normed by stationary distribution		5.3	23.8	11.0	11.3	21.3
commute time		5.2	15.5	**33.1**	*17.1*	23.4
commute time—normed by stationary distribution		5.3	16.1	11.0	11.3	16.3
rooted PageRank	$\alpha = 0.01$	*10.8*	*28.0*	**33.1**	*18.7*	*29.2*
	$\alpha = 0.05$	*13.8*	*39.9*	**35.3**	*24.6*	*41.3*
	$\alpha = 0.15$	*16.6*	**41.1**	**27.2**	**27.6**	*42.6*
	$\alpha = 0.30$	*17.1*	**42.3**	*25.0*	**29.9**	*46.8*
	$\alpha = 0.50$	*16.8*	**41.1**	*24.3*	**30.7**	*46.8*
Katz (weighted)	$\beta = 0.05$	3.0	21.4	19.9	2.4	12.9
	$\beta = 0.005$	*13.4*	**54.8**	**30.1**	*24.0*	**52.2**
	$\beta = 0.0005$	*14.5*	**54.2**	**30.1**	**32.6**	**51.8**
Katz (unweighted)	$\beta = 0.05$	*10.9*	**41.7**	**37.5**	*18.7*	**48.0**
	$\beta = 0.005$	*16.8*	**41.7**	**37.5**	*24.2*	**49.7**
	$\beta = 0.0005$	*16.8*	**41.7**	**37.5**	*24.9*	**49.7**

Figure 3: Performance of various predictors on the link prediction task defined in Section 2. For each predictor and each arXiv section, the given number specifies the factor improvement over random prediction. Two predictors in particular are used as baselines for comparison: graph distance and common neighbors (see Section 3 for definitions of these). Italicized entries have performance at least as good as the graph distance predictor; bold entries are at least as good as the common neighbors predictor. See also Figure 4.

predictor		astro-ph	cond-mat	gr-qc	hep-ph	hep-th
probability that a random prediction is correct		0.475%	0.147%	0.341%	0.207%	0.153%
graph distance (all distance-two pairs)		*9.6*	*25.3*	*21.4*	*12.2*	*29.2*
common neighbors		**18.0**	**41.1**	**27.2**	**27.0**	**47.2**
Low-rank approximation:	rank = 1024	*15.2*	**54.2**	**29.4**	**34.9**	**50.1**
Inner product	rank = 256	*14.6*	**47.1**	**29.4**	**32.4**	**47.2**
	rank = 64	*13.0*	**44.7**	**27.2**	**30.8**	**47.6**
	rank = 16	*10.1*	21.4	**31.6**	**27.9**	*35.5*
	rank = 4	8.8	15.5	**42.6**	*19.6*	23.0
	rank = 1	6.9	6.0	**44.9**	*17.7*	14.6
Low-rank approximation:	rank = 1024	8.2	16.7	6.6	*18.6*	21.7
Matrix entry	rank = 256	*15.4*	*36.3*	8.1	*26.2*	*37.6*
	rank = 64	*13.8*	**46.5**	16.9	**28.1**	*40.9*
	rank = 16	9.1	21.4	*26.5*	*23.1*	*34.2*
	rank = 4	8.8	15.5	**39.7**	*20.0*	22.5
	rank = 1	6.9	6.0	**44.9**	*17.7*	14.6
Low-rank approximation:	rank = 1024	*11.4*	*27.4*	**30.1**	**27.1**	*32.1*
Katz ($\beta = 0.005$)	rank = 256	*15.4*	**42.3**	11.0	**34.3**	*38.8*
	rank = 64	*13.1*	**45.3**	19.1	**32.3**	*41.3*
	rank = 16	9.2	21.4	**27.2**	*24.9*	*35.1*
	rank = 4	7.0	15.5	**41.2**	*19.7*	23.0
	rank = 1	0.4	6.0	**44.9**	*17.7*	14.6
unseen bigrams	common neighbors, $\delta = 8$	*13.5*	*36.9*	**30.1**	*15.6*	**47.2**
(weighted)	common neighbors, $\delta = 16$	*13.4*	*39.9*	**39.0**	*18.6*	**48.8**
	Katz ($\beta = 0.005$), $\delta = 8$	*16.9*	*38.1*	*25.0*	*24.2*	**51.3**
	Katz ($\beta = 0.005$), $\delta = 16$	*16.5*	*39.9*	**35.3**	*24.8*	**50.9**
unseen bigrams	common neighbors, $\delta = 8$	*14.2*	*40.5*	**27.9**	*22.3*	*39.7*
(unweighted)	common neighbors, $\delta = 16$	*15.3*	*39.3*	**42.6**	*22.1*	*42.6*
	Katz ($\beta = 0.005$), $\delta = 8$	*13.1*	*36.9*	**32.4**	*21.7*	*38.0*
	Katz ($\beta = 0.005$), $\delta = 16$	*10.3*	*29.8*	**41.9**	*12.2*	*38.0*
clustering:	$\rho = 0.10$	7.4	*37.5*	**47.1**	**33.0**	*38.0*
Katz ($\beta_1 = 0.001, \beta_2 = 0.1$)	$\rho = 0.15$	*12.0*	**46.5**	**47.1**	*21.1*	*44.2*
	$\rho = 0.20$	4.6	*34.5*	19.9	*21.2*	*35.9*
	$\rho = 0.25$	3.3	*27.4*	20.6	*19.5*	17.5

Figure 4: Performance of various meta-approaches on the link prediction task defined in Section 2. As before, for each predictor and each arXiv section, the given number specifies the factor improvement over random prediction. See Figure 3.

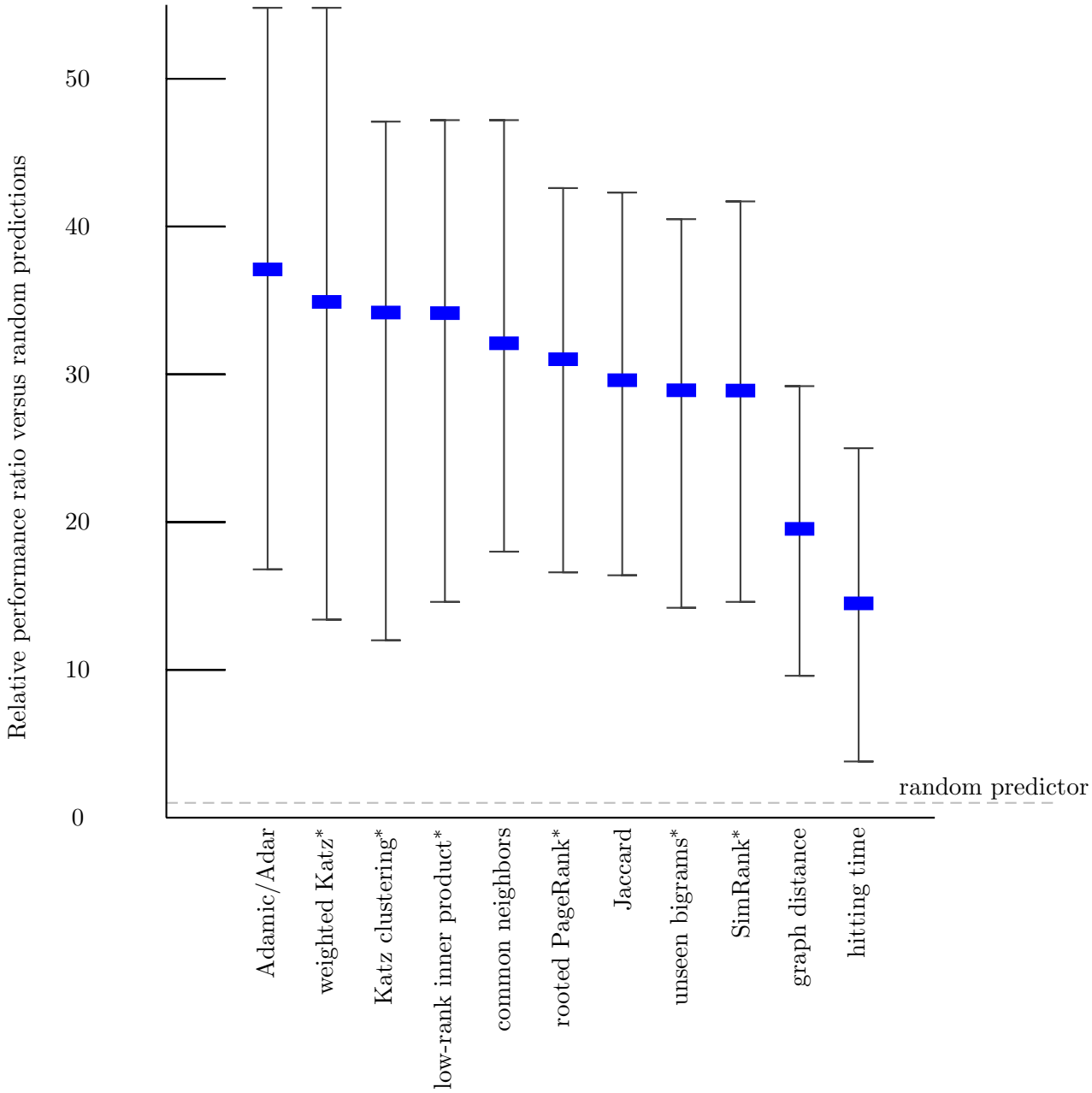

Figure 5: Relative average performance of various predictors versus random predictions. The value shown is the average ratio over the five datasets of the given predictor's performance versus the random predictor's performance. The error bars indicate the minimum and maximum of this ratio over the five datasets. The parameters for the starred predictors are: (1) for weighted Katz, $\beta = 0.005$; (2) for Katz clustering, $\beta_1 = 0.001, \rho = 0.15, \beta_2 = 0.1$; (3) for low-rank inner product, $\mathsf{rank} = 256$; (4) for rooted Pagerank, $\alpha = 0.15$; (5) for unseen bigrams, unweighted common neighbors with $\delta = 8$; and (6) for SimRank, $\gamma = 0.8$.

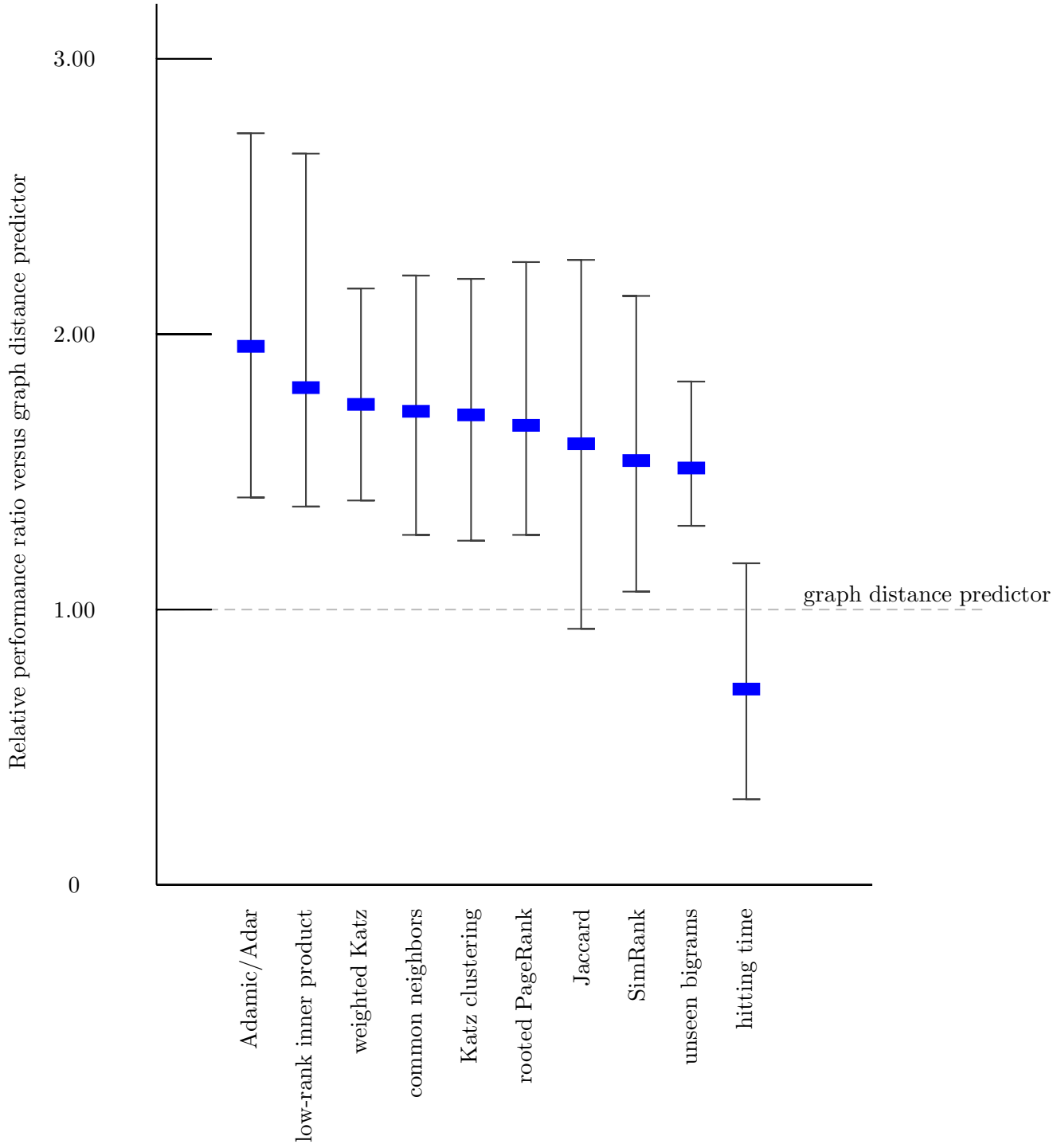

Figure 6: Relative average performance of various predictors versus the graph distance predictor. The plotted value shows the average taken over the five datasets of the ratio of the performance of the given predictor versus the graph distance predictor; the error bars indicate the range of this ratio over the five datasets. All parameter settings are as in Figure 5.

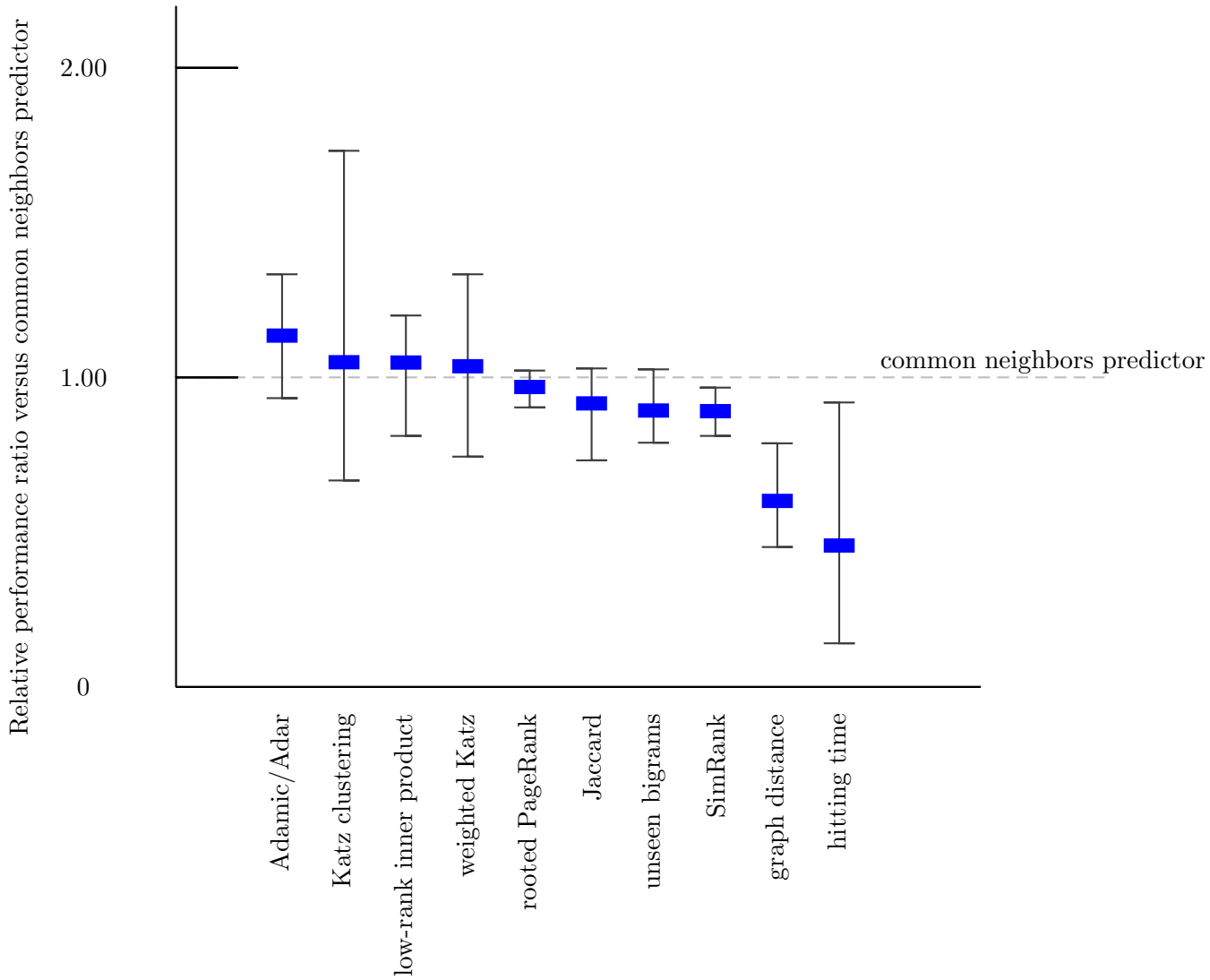

Figure 7: Relative average performance of various predictors versus the common neighbors predictor, as in Figure 6. Error bars display the range of the performance ratio of the given predictor versus common neighbors over the five datasets; the displayed value gives the average ratio. Parameter settings are as in Figure 5.

	Adamic/Adar	Katz clustering	common neighbors	hitting time	Jaccard's coefficient	weighted Katz	low-rank inner product	rooted Pagerank	SimRank	unseen bigrams
Adamic/Adar	1150	638	520	193	442	1011	905	528	372	486
Katz clustering		1150	411	182	285	630	623	347	245	389
common neighbors			1150	135	506	494	467	305	332	489
hitting time				1150	87	191	192	247	130	156
Jaccard's coefficient					1150	414	382	504	845	458
weighted Katz						1150	1013	488	344	474
low-rank inner product							1150	453	320	448
rooted Pagerank								1150	678	461
SimRank									1150	423
unseen bigrams										1150

Figure 8: The number of common predictions made by various predictors on the `cond-mat` dataset, out of 1150 predictions. Parameter settings are as in Figure 5.

arXiv sections, a variant of Katz achieves the best performance. Some of the very simple measures also perform surprisingly well, including common neighbors and the Adamic/Adar measure.

Similarities among the predictors and the datasets. Not surprisingly, there is significant overlap in the predictions made by the various methods. In Figure 8, we show the number of common predictions made by ten of the most successful measures on the `cond-mat` graph. We see that Katz, low-rank inner product, and Adamic/Adar are quite similar in their predictions, as are (to a somewhat lesser extent) rooted PageRank, SimRank, and Jaccard. Hitting time is remarkably unlike any of the other nine in its predictions, despite its reasonable performance. The number of common *correct* predictions shows qualitatively similar behavior; see Figure 9. It would interesting to understand the generality of these overlap phenomena, especially since certain of the large overlaps do not seem to follow obviously from the definitions of the measures.

It is harder to quantify the relationships among the datasets, but this is a very interesting issue as well. One perspective is provided by the methods based on low-rank approximation: on four of the datasets, their performance tends to be best at an intermediate rank, while on `gr-qc` they perform best at rank 1. This suggests a sense in which the collaborations in `gr-qc` have a much "simpler" structure than in the other four. One also observes the apparent importance of node degree in the `hep-ph` collaborations: the preferential attachment predictor—which considers only the number (and not the identity) of a scientist's co-authors—does uncharacteristically well on this dataset, outperforming the basic graph distance predictor. Finally, it would be interesting to make precise a sense in which `astro-ph` is a "difficult" dataset, given the low performance of all methods relative to random, and the fact that none beats simple ranking by common neighbors. We will explore this issue further below when we consider collaboration data drawn from other fields.

	Adamic/Adar	Katz clustering	common neighbors	hitting time	Jaccard's coefficient	weighted Katz	low-rank inner product	rooted Pagerank	SimRank	unseen bigrams
Adamic/Adar	92	65	53	22	43	87	72	44	36	49
Katz clustering		78	41	20	29	66	60	31	22	37
common neighbors			69	13	43	52	43	27	26	40
hitting time				40	8	22	19	17	9	15
Jaccard's coefficient					71	41	32	39	51	43
weighted Katz						92	75	44	32	51
low-rank inner product							79	39	26	46
rooted Pagerank								69	48	39
SimRank									66	34
unseen bigrams										68

Figure 9: The number of *correct* common predictions made by various predictors on the `cond-mat` dataset, out of 1150 predictions. The diagonal entries indicate the number of correct predictions for each predictor. Parameter settings are as in Figure 5.

Small worlds. It is reassuring that even the basic graph distance predictor handily outperforms random predictions, but this measure has severe limitations. Extensive research has been devoted to understanding the so-called *small world problem* in collaboration networks—i.e., accounting for the existence of short paths connecting virtually every pair of scientists [18]. This property is normally viewed as a vital fact about the scientific community (new ideas spread quickly, and every discipline interacts with and gains from other fields) but in the context of our prediction task, we come to a different conclusion: the small world problem is really a problem. The shortest path between two scientists in wholly unrelated disciplines is often very short (and very tenuous). To take one particular (but not atypical) example, the developmental psychologist Jean Piaget has as small an Erdös Number—three [4]—as most mathematicians and computer scientists. Overall, the basic graph distance predictor is not competitive with most of the other approaches studied; our most successful link predictors can be viewed as using measures of proximity that are robust to the few edges that result from rare collaborations between fields.

Restricting to distance three. The small world problem suggests that there are many pairs with graph distance two that will not collaborate, but we also observe the dual problem: many pairs that collaborate are at distance larger than two. Between 71% (`hep-ph`) and 83% (`cond-mat`) of new edges form between pairs at distance three or greater; see Figure 10.

Since most new collaborations are not at distance two, we are also interested in how well our predictors perform when we disregard all distance-two pairs. Clearly, nodes at distance greater than two have no neighbors in common, and hence this task essentially rules out the use of methods based on common neighbors. The performance of the other measures is shown in Figure 11. The graph

	astro-ph	cond-mat	gr-qc	hep-ph	hep-th
# pairs at distance two	33862	5145	935	37687	7545
# new collaborations at distance two	1533	190	68	945	335
# new collaborations	5751	1150	400	3294	1576

Figure 10: Relationship between new collaborations and graph distance.

distance predictor (i.e., predicting all distance-three pairs) performs between three and eight times random, and is consistently beaten by virtually all of the predictors: SimRank, rooted PageRank, Katz, and the low-rank approximation and unseen bigram techniques. The unweighted Katz and unseen bigram predictors have the best performance (as high as about 30 times random, on `gr-qc`), followed closely by weighted Katz, SimRank, and rooted PageRank.

The breadth of the data. We also have considered three other datasets: (1) the proceedings of the theoretical computer science conferences Symposium on the Theory of Computing (STOC) and Foundations of Computer Science (FOCS), (2) the papers found in the Citeseer (`www.citeseer.com`) online database, which finds papers by crawling the web for any files in postscript form, and (3) all five of the arXiv sections merged into one. Consider the performance of the common neighbor predictor versus random on these datasets:

STOC/FOCS	arXiv sections	all arXiv's	Citeseer
6.1	18.0—41.1	71.2	147.0

Performance versus random swells dramatically as the topical focus of our data set widens. That is, when we consider a more diverse collection of scientists, it is fundamentally easier to group scientists into fields of study (and outperform random predictions, which will usually make guesses between fields). When we consider a sufficiently narrow set of researchers—e.g., STOC/FOCS—almost any author can collaborate with almost any other author, and there seems to a strong random component to new collaborations. (In extensive experiments on the STOC/FOCS data, we could not beat random guessing by a factor of more than about seven.) It is an interesting challenge to formalize the sense in which the STOC/FOCS collaborations are truly intractable to predict—i.e., to what extent information about new collaborations is simply not present in the old collaboration data.

Future directions. While the predictors we have discussed perform reasonably well, even the best (Katz clustering on `gr-qc`) is correct on only about 16% of its predictions. There is clearly much room for improvement in performance on this task, and finding ways to take better advantage of the information in the training data is an interesting open question. Another issue is to improve the efficiency of the proximity-based methods on very large networks; fast algorithms for approximating the distribution of node-to-node distances may be one approach [21].

The graph G_{collab} is a lossy representation of the data; we can also consider a bipartite collaboration graph B_{collab}, with a vertex for every author and paper, and an edge connecting each paper to each of its authors. The bipartite graph contains more information than G_{collab}, so we may hope that predictors can use it to improve performance. The size of B_{collab} is much larger than G_{collab}, making experiments prohibitive, but we have tried using the SimRank and Katz predictors on smaller datasets (`gr-qc`, or shorter training periods). Their performance does not seem to improve, but perhaps other predictors can fruitfully exploit the additional information in B_{collab}.

predictor		astro-ph	cond-mat	gr-qc	hep-ph	hep-th
graph distance (all distance-three pairs)		3.1	5.5	8.4	3.8	8.4
preferential attachment		3.2	2.6	8.6	4.7	1.4
SimRank	$\gamma = 0.8$	6.0	14.4	10.6	7.7	22.0
hitting time		4.4	10.2	13.7	4.5	4.7
hitting time—normed by stationary distribution		2.0	2.5	0.0	2.6	6.7
commute time		4.0	5.9	21.1	6.0	6.7
commute time—normed by stationary distribution		2.6	0.8	1.1	4.8	4.7
rooted PageRank	$\alpha = 0.01$	4.6	12.7	21.1	6.5	12.7
	$\alpha = 0.05$	5.4	13.6	21.1	8.8	16.6
	$\alpha = 0.15$	5.4	11.9	18.0	11.1	20.0
	$\alpha = 0.30$	5.9	13.6	8.5	11.9	20.0
	$\alpha = 0.50$	6.4	15.2	7.4	13.1	20.0
Katz (weighted)	$\beta = 0.05$	1.5	5.9	11.6	2.3	2.7
	$\beta = 0.005$	5.8	14.4	28.5	4.3	12.7
	$\beta = 0.0005$	6.3	13.6	27.5	4.3	12.7
Katz (unweighted)	$\beta = 0.05$	2.4	12.7	30.6	9.1	12.7
	$\beta = 0.005$	9.2	11.9	30.6	5.1	18.0
	$\beta = 0.0005$	9.3	11.9	30.6	5.1	18.0
Low-rank approximation:	rank = 1024	2.3	2.5	9.5	4.0	6.0
Inner product	rank = 256	4.8	5.9	5.3	10.2	10.7
	rank = 64	3.9	12.7	5.3	7.1	11.3
	rank = 16	5.4	6.8	6.3	6.8	15.3
	rank = 4	5.4	6.8	32.8	2.0	4.7
	rank = 1	6.1	2.5	32.8	4.3	8.0
Low-rank approximation:	rank = 1024	4.1	6.8	6.3	6.2	13.3
Matrix entry	rank = 256	3.8	8.5	3.2	8.5	20.0
	rank = 64	3.0	11.9	2.1	4.0	10.0
	rank = 16	4.6	8.5	4.2	6.0	16.6
	rank = 4	5.2	6.8	27.5	2.0	4.7
	rank = 1	6.1	2.5	32.8	4.3	8.0
Low-rank approximation:	rank = 1024	4.3	6.8	28.5	6.2	13.3
Katz ($\beta = 0.005$)	rank = 256	3.6	8.5	3.2	8.5	20.6
	rank = 64	2.9	11.9	2.1	4.3	10.7
	rank = 16	5.1	8.5	5.3	6.0	16.0
	rank = 4	5.5	6.8	28.5	2.0	4.7
	rank = 1	0.3	2.5	32.8	4.3	8.0
unseen bigrams	common neighbors, $\delta = 8$	5.8	6.8	14.8	4.3	24.0
(weighted)	common neighbors, $\delta = 16$	7.9	9.3	28.5	5.1	19.3
	Katz ($\beta = 0.005$), $\delta = 8$	5.2	10.2	22.2	2.8	18.0
	Katz ($\beta = 0.005$), $\delta = 16$	6.6	10.2	29.6	3.7	15.3
unseen bigrams	common neighbors, $\delta = 8$	5.6	5.1	13.7	4.5	21.3
(unweighted)	common neighbors, $\delta = 16$	6.4	8.5	25.4	4.8	22.0
	Katz ($\beta = 0.005$), $\delta = 8$	4.2	7.6	22.2	2.0	17.3
	Katz ($\beta = 0.005$), $\delta = 16$	4.3	4.2	28.5	3.1	16.6
clustering:	$\rho = 0.10$	3.5	4.2	31.7	7.1	8.7
Katz ($\beta_1 = 0.001, \beta_2 = 0.1$)	$\rho = 0.15$	4.8	4.2	32.8	7.7	6.7
	$\rho = 0.20$	2.5	5.9	7.4	4.5	8.0
	$\rho = 0.25$	2.1	11.9	6.3	6.8	5.3

Figure 11: The Distance-3 Task: performance of predictors only on edges in E_{new} for which the endpoints were at distance three or more in G_{collab}. Methods based on common neighbors are not appropriate for this task. See Section 4.

Similarly, our experiments treat all training period collaborations equally. Perhaps one can improve performance by treating more recent collaborations as more important than older ones. One could also tune the parameters of the Katz predictor, e.g., by dividing the training set into temporal segments, training β on the beginning, and then using the end of the training set to make final predictions.

Finally, there has been relevant work in the machine learning community on *estimating distribution support*: given samples from an unknown probability distribution P, we must find a "simple" set S so that $\mathsf{Pr}_{x \sim P}[x \notin S] < \varepsilon$ [25]. We can view training period collaborations as samples drawn from a probability distribution on pairs of scientists; our goal is to approximate the set of pairs that have positive probability of collaborating. It is an open question whether these techniques can be fruitfully applied to the link prediction problem.

Acknowledgements. We thank Jon Herzog, Tommi Jaakkola, Lillian Lee, Frank McSherry, and Grant Wang for helpful discussions and comments on earlier drafts of this paper. We thank Paul Ginsparg for generously providing the bibliographic data from the arXiv.

References

[1] Lada A. Adamic and Eytan Adar. Friends and neighbors on the web. *Social Networks*, 25(3):211–230, July 2003.

[2] A. L. Barabasi, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, and T. Vicsek. Evolution of the social network of scientific collaboration. *Physica A*, 311(3–4):590–614, 2002.

[3] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[4] Rodrigo De Castro and Jerrold W. Grossman. Famous trails to Paul Erdös. *Mathematical Intelligencer*, 21(3):51–63, 1999.

[5] Jörn Davidsen, Holger Ebel, and Stefan Bornholdt. Emergence of a small world from local interactions: Modeling acquaintance networks. *Physical Review Letters*, 88(128701), 2002.

[6] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[7] Ute Essen and Volker Steinbiss. Cooccurrence smoothing for stochastic language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 161–164, 1992.

[8] Debra S. Goldberg and Frederick P. Roth. Assessing experimentally derived interactions in a small world. In *Proceedings of the National Academy of Sciences USA*, volume 100, pages 4372–4376, April 2003.

[9] Jerrold W. Grossman. The evolution of the mathematical research collaboration graph. In *Proceedings of the Southeast Conference on Combinatorics, Graph Theory, and Computing*, March 2002.

[10] Glen Jeh and Jennifer Widom. SimRank: A measure of structural-context similarity. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, July 2002.

[11] Emily M. Jin, Michelle Girvan, and M. E. J. Newman. The structure of growing social networks. *Physical Review Letters E*, 64(046132), 2001.

[12] Leo Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, March 1953.

[13] H. Kautz, B. Selman, and M. Shah. ReferralWeb: Combining social networks and collaborative filtering. *Communications of the ACM*, 30(3), March 1997.

[14] Valdis Krebs. Mapping networks of terrorist cells. *Connections*, 24(3):43–52, Winter 2002.

[15] Lillian Lee. Measures of distributional similarity. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 25–32, 1999.

[16] Michael Mitzenmacher. A brief history of lognormal and power law distributions. In *Proceedings of the Allerton Conference on Communication, Control, and Computing*, pages 182–191, 2001.

[17] M. E. J. Newman. Clustering and preferential attachment in growing networks. *Physical Review Letters E*, 64(025102), 2001.

[18] M. E. J. Newman. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences USA*, 98:404–409, 2001.

[19] M. E. J. Newman. The structure and function of networks. *Computer Physics Communications*, 147:40–45, 2002.

[20] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

[21] Christopher Palmer, Phillip Gibbons, and Christos Faloutsos. ANF: A fast and scalable tool for data mining in massive graphs. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Jul 2002.

[22] A. Popescul and L. Ungar. Statistical relational learning for link prediction. In *Workshop on Learning Statistical Models from Relational Data at the International Joint Conference on Artificial Intelligence*, 2003.

[23] P. Raghavan. Social networks: From the web to the enterprise. *IEEE Internet Computing*, 6(1):91–94, January/February 2002.

[24] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

[25] Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a high-dimensional distribution. Technical Report MSR-TR-99-87, Microsoft Research, 1999.

[26] Ben Taskar, Ming-Fai Wong, Pieter Abbeel, and Daphne Koller. Link prediction in relational data. In *Proceedings of Neural Information Processing Systems*, 2004. To appear.

[27] Duncan J. Watts. *Small Worlds*. Princeton University Press, 1999.

[28] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.