Pairwise Link Prediction

Huda Nassar
Computer Science Department
Purdue University
West Lafayette, USA
hnassar@purdue.edu

Austin R. Benson
Computer Science Department
Cornell University
Ithaca, USA
arb@cs.cornell.edu

David F. Gleich
Computer Science Department
Purdue University
West Lafayette, USA
dgleich@purdue.edu

***Abstract*—Link prediction is a common problem in network science that transects many disciplines. The goal is to forecast the appearance of new links or to find links missing in the network. Typical methods for link prediction use the topology of the network to predict the most likely future or missing connections between a pair of nodes. However, network evolution is often mediated by higher-order structures involving more than pairs of nodes; for example, cliques on three nodes (also called triangles) are key to the structure of social networks, but the standard link prediction framework does not directly predict these structures. To address this gap, we propose a new link prediction task called "pairwise link prediction" that directly targets the prediction of new triangles, where one is tasked with finding which nodes are most likely to form a triangle with a given edge. We develop two PageRank-based methods for our pairwise link prediction problem and make natural extensions to existing link prediction methods. Our experiments on a variety of networks show that diffusion based methods are less sensitive to the type of graphs used and more consistent in their results. We also show how our pairwise link prediction framework can be used to get better predictions within the context of standard link prediction evaluation.**

***Index Terms*—link prediction, higher order methods, PageRank**

I. Introduction

Networks are a standard tool for data analysis in which links between data points are the primary object of study. A fundamental problem in network analysis is *link prediction* [1], [2], which is typically formulated as a problem of identifying pairs of nodes that will either form a link in the future (when viewing the network as evolving over time) or whose connection is missing from the data [3]. The link prediction problem has applications in a variety of domains. For instance, in online social networks of friendships, predicting that two people will form a connection can be used for friendship recommendation [4]. Similarly, predicting new links between users and items on platforms such as Amazon and Netflix can be used for product recommendation [5]. And in biology, link prediction is used to identify novel interactions between genes, diseases, and proteins within interaction networks [6].

In the settings above, the link prediction problem is oriented around—and evaluated in terms of—the identification of *pairs of nodes* that are likely to be connected. However, there is mounting evidence that the organization and evolution of networks is centered around higher-order interactions involving more than two nodes [7]–[11]. In the case of social networks, triangles (a clique on three nodes) are extremely common due to various sociological mechanisms driving triadic closure [12]–[15]. Methods for link prediction are indeed motivated by these ideas. For instance, the Jaccard similarity between the sets of neighbors of two nodes—a common heuristic for link prediction [1]—measures the number of triangles that would be created if the two nodes are linked, normalized by the total number of neighbors of the two nodes. Still, methods such as Jaccard similarity are used to make predictions on *pairs of nodes*, rather than a prediction on the appearance of the higher-order structures directly.

Here, we develop a framework for directly predicting the appearance of a higher-order structure. We focus on the case of triangles, which is one of the simplest higher-order structures while also being critical to social network analysis. Again, classical link prediction is centered around the following question: given a node u in the network, which nodes are likely to link to u? This scenario is illustrated in Figure 1A. Our framing of the problem is similar, but we instead ask the following: given an edge (u, v) in the network, which nodes are likely to connect to both u and v? We call this the *pairwise link prediction problem*, and it is illustrated in Figure 1B. There are several scenarios where the pairwise link prediction problem is natural, such as recommending a new friend to a couple on an online social network, recommending a movie to a couple in a video site, or predicting an effective drug given a disease-gene pair.

We devise two new algorithms for the pairwise link prediction problem. The first is based on a variant of seeded (personalized) PageRank that uses multiple seeds, namely, one seed at each end point of the edge for which we are trying to predict new triadic connections. The second is based on a PageRank-like iteration that puts more weight on edges that participate in many triangles. In this sense, the method reinforces triangles, and we call the method "Triangle Reinforced PageRank" (TRPR). We compare these algorithms to natural

ASONAM '19, August 27-30, 2019, Vancouver, Canada

ACM ISBN 978-1-4503-6868-1/19/08?/$15.00
http://dx.doi.org/10.1145/3341161.3342897

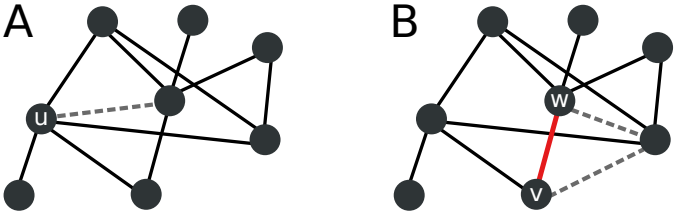

Fig. 1. **(A)** In standard link prediction, we are tasked with finding nodes that are likely to link to a given node u. **(B)** In this paper, we study pairwise link prediction, where we are tasked with finding nodes that are likely to form a triangle with a given edge (v, w).

extensions of local similarity measures that are common in link prediction, such as Jaccard similarity [1], Adamic-Adar similarity [16], and preferential attachment [17].

For a given edge, each of the above methods produces a score for the remaining nodes in the graph. We use the ordering of these scores to measure the area under the ROC curve with respect to a held-out test set. We find that our proposed pairseeded PageRank and TRPR methods outperform the baseline measures based on local neighborhood information on a number of synthetic benchmark and real-world datasets. For instance, on predicting future links in a temporal graph, our methods had median AUC scores of 0.93 compared with 0.76 for the baseline methods. Based on this success of our PageRank-based methods for pairwise link prediction, we then go back and adapt them for the classical link prediction problem. We find that these adapted methods out-perform traditional seeded PageRank on a number of real-world datasets with average AUC increases of up to 0.28 in predicting missing drug interactions.

II. Background and Related Work

We now briefly review some related work in link prediction and higher-order structure. As part of this, we will go over methods that we will generalize in the next section for the pairwise link prediction problem. All of these methods assign some similarity score between pairs of nodes, where a larger similarity is indicative of pairs that are likely to connect. For notation, we use $\Gamma(u)$ to denote the set of neighbors of node u in the graph.

Local methods. Several approaches to link prediction are based on local information in the graph, namely a score is assigned to a pair of nodes u and v based on their 1-hop neighborhoods $\Gamma(u)$ and $\Gamma(v)$. One approach that falls under this category stems from the idea that as $|\Gamma(u) \cap \Gamma(v)|$ increases, the chance that u and v are connected also increases [17]. Here, $|\Gamma(u) \cap \Gamma(v)|$ is the number of triangles that would be formed if u and v were connected. Often, this number is normalized by the size of the neighborhoods, which gives rise to the Jaccard similarity between two nodes u and v:

$$\frac{|\Gamma(u) \cap \Gamma(v)|}{|\Gamma(u) \cup \Gamma(v)|}.$$

The Adamic–Adar similarity measure [16] is a local score that assigns similarity between two nodes based on how important their common neighbors are, where importance is measured by the inverse log degree of a node. Formally, the Adamic–Adar similarity measure between nodes u and v is:

$$\sum_{z \in \Gamma(u) \cap \Gamma(v)} \frac{1}{\log(|\Gamma(z)|)}.$$

A third local method is based on preferential attachment, where nodes are more likely to connect to *established* nodes in the network, and, *established* nodes have a higher chance to connect to each other [17], [18]. Using degree as a proxy for how established a node is, the preferential attachment score between nodes u and v is:

$$|\Gamma(u)| \cdot |\Gamma(v)|.$$

Global methods. Another set of approaches for link prediction are based on aggregating (weighted or normalized) path counts of varying lengths. In contrast to the local methods described above, these methods use global information about the entire network. For example, the Katz similarity counts the number of paths between two nodes, weighting paths of length-k by β^k [1], [19]. Another class of global methods are methods based on conservative diffusions such as PageRank [20]. Such diffusion methods are typically *seeded* by a particular node u, and the similarity of u to all other nodes is given by the amount of "mass" that diffuses to each other node. We will make use of PageRank-like methods in the next section.

Higher-order structure. Since a network encodes pairwise relationships (edges) between elements (nodes), the link prediction problem is natural in many cases. Nevertheless, recent studies have shown that networks evolve through higher-order interactions, i.e., much of the structure in evolving networks involves interactions between more than just two nodes [10]. Moreover, random graph models constructed from distributions of triangles have shown to be good fits for real-world data [21], providing additional evidence that triadic relationships are important to the assembly of networks.

III. Methods

We propose several methods for the pairwise link prediction problem. First, we extend the three local metrics described above to measure node-edge similarity—these methods will serve as our baseline metrics. After, we propose two diffusion-based methods akin to seeded PageRank.

A. Local similarity measures for pairwise prediction

Our goal here is to extend common local methods for link prediction to the scenario of pairwise link prediction. In other words, instead of computing similarity between nodes, we now compute similarity between an edge and a node. To do this, we simply replace the neighborhood of one node with the neighborhood of an edge. This requires that we specify what the neighborhood of an edge (u, v) should capture. We define:

$$\begin{aligned}\Gamma((u, v)) &= \{\text{node } z \mid z \text{ forms a triangle with } (u, v)\} \\ &= \Gamma(u) \cap \Gamma(v).\end{aligned}$$

Note that this is different from the boundary of a set of vertices in the graph that is often used to define the size of a cut, which—for an edge—would correspond to the union of neighborhoods. Our choice here preserves the intuition behind the link prediction methods given below in the context of the pairwise prediction problem because we want strong relationships with both u and v. We explore other types of neighborhoods in an extended version of this paper [22].

Using the substitution gives us three similarity measures that will serve as our baseline methods:

- Jaccard Similarity.

$$JS(w,(u,v)) = \frac{|\Gamma(w) \cap \Gamma((u,v))|}{|\Gamma(w) \cup \Gamma((u,v))|}$$

- Adamic–Adar.

$$AA(w,(u,v)) = \sum_{z \in \Gamma(w) \cap \Gamma((u,v))} \frac{1}{\log|\Gamma(z)|}$$

- Preferential Attachment.

$$PA(w,(u,v)) = |\Gamma(w)| \cdot |\Gamma((u,v))|$$

Next, we develop two new methods for pairwise link prediction based on seeded PageRank.

B. Pair-seeded PageRank

Seeded PageRank is a foundational concept in network analysis that models a flow of information in a network to predict links and communities on a network [23], [24]. Seeded PageRank models information flow from the seed node to other nodes in the network via a Markov chain, and the stationary distribution of the chain provides the scores on the nodes. A high score on a node is a signal that the node should be connected to the seed node. More formally, let $\boldsymbol{A}$ be the symmetric adjacency matrix of an undirected graph, and let $\boldsymbol{P}$ be the column stochastic matrix of a random walk on that graph. Specifically, $P(i,j) = A(i,j)/|\Gamma(j)|$. Let u be the seed node. Then the seeded PageRank scores are entries of the solution vector $\mathbf{x}$ to the linear system $(\boldsymbol{I} - \alpha\boldsymbol{P})\mathbf{x} = (1-\alpha)\mathbf{e}_u$.

Here, $\mathbf{e}_u$ is the vector of all zeros, except at index u, where $\mathbf{e}_u(u) = 1$ (i.e., $\mathbf{e}_u$ is the indicator vector on node u) and α is the PageRank damping parameter. The entries of $\mathbf{x}$ provide similarities between node u and the other nodes and thus can be used for standard link prediction.

In the same way seeded PageRank predicts the relevance of other nodes in the network to a single seed node, we propose *pair-seeded PageRank* to predict the relevance of nodes to a single edge; with these similarities, we are able to make predictions for the pairwise link prediction problem. For a given edge (u,v), pair-seeded PageRank solves the following linear system:

$$(\boldsymbol{I} - \alpha\boldsymbol{P})\mathbf{x} = (1-\alpha)\mathbf{e}_{u,v}.$$

In this case, $\mathbf{e}_{u,v}$ is the vector of all zeros, except at indices u and v, where $\mathbf{e}_{u,v}(u) = \mathbf{e}_{u,v}(v) = 1/2$. The solution $\mathbf{x}$ can be interpreted as the similarity of each node to the edge (u,v).

We now note that pair-seeded PageRank is equivalent to the sum of single-seeded PageRank on each of the nodes, up to a scalar multiple. This follows quickly from linearity of the PageRank problem. To see this, let $\mathbf{x}_u$ and $\mathbf{x}_v$ be the seeded PageRank solutions corresponding to nodes u and v respectively. Then,

$$(\boldsymbol{I} - \alpha\boldsymbol{P})\mathbf{x}_u = (1-\alpha)\mathbf{e}_u$$
$$(\boldsymbol{I} - \alpha\boldsymbol{P})\mathbf{x}_v = (1-\alpha)\mathbf{e}_v.$$

Adding the above two equations yields

$$(\boldsymbol{I} - \alpha\boldsymbol{P})(\mathbf{x}_u + \mathbf{x}_v) = (1-\alpha)(\mathbf{e}_u + \mathbf{e}_v)$$
$$(\boldsymbol{I} - \alpha\boldsymbol{P})(\mathbf{x}_u + \mathbf{x}_v) = (1-\alpha)(2\mathbf{e}_{u,v})$$
$$\frac{1}{2}(\boldsymbol{I} - \alpha\boldsymbol{P})(\mathbf{x}_u + \mathbf{x}_v) = (1-\alpha)\mathbf{e}_{u,v}$$
$$(\boldsymbol{I} - \alpha\boldsymbol{P})\mathbf{x} = (1-\alpha)\mathbf{e}_{u,v}.$$

Hence, $2\mathbf{x} = \mathbf{x}_u + \mathbf{x}_v$, and the pair-seeded PageRank solution is equivalent to the summation of the single seeded PageRank equations, up to scaling. Indeed, this is a useful observation as there are many systems designed to estimate large seeded PageRank values for single-seeds by using highly scalable random walk methods [25]. Thus this technique could be used wherever a PageRank-style prediction is already employed.

C. Triangle Reinforced PageRank (TRPR)

We now propose a PageRank-like method that uses a weighting scheme on edges based on the number of triangles that contains each edge, which we call *Triangle Reinforced PageRank* (TRPR). For an unweighted graph, the PageRank solution is highly affected by the degree of nodes in the network. Here, we *reinforce* the influence of triangles by giving edges participating in many triangles a higher weight. Figure 2 presents a motivating example for the usefulness of reinforcing triangles.

To develop our TRPR method, we first introduce a tensor $\underline{\boldsymbol{T}}$, that encodes all triangles in a network:

$$\underline{\boldsymbol{T}}(i,j,k) = \begin{cases} 1 & \text{if } (i,j,k) \text{ is a triangle} \\ 0 & \text{otherwise.} \end{cases}$$

Again, in our derivation, we assume that the graph is undirected so that $\underline{\boldsymbol{T}}$ is fully symmetric in all permutations of indices.

A typical way to solve the PageRank linear system is the power method. With TRPR, we modify the power method by adding a step that redistributes the weights in the network. Specifically, we compute the matrix $\hat{\boldsymbol{X}} = \underline{\boldsymbol{T}}[\mathbf{x}]$, where $\hat{\boldsymbol{X}}(i,j) = \sum_k \underline{\boldsymbol{T}}(i,j,k)\mathbf{x}(k)$, which measures the relevance of edge (i,j) to the distribution of node scores in the vector $\mathbf{x}$. We then run an iteration of the power method on a weighted adjacency matrix $\boldsymbol{X} = \hat{\boldsymbol{X}} + \boldsymbol{A}$, where the columns are renormalized to make the matrix column stochastic. Algorithm 1 shows the idealized algorithm.

TRPR can be implemented efficiently. Although TRPR involves the tensor $\underline{\boldsymbol{T}}$, we do not need to form it explicitly, and we show an alternative derivation here. We first unwrap one

iteration of TRPR. Let $\boldsymbol{A}_i = \underline{\boldsymbol{T}}[\mathbf{x}_{i-1}] + \boldsymbol{A}$. Then, at iteration i, we can translate $\mathbf{x}_i = \alpha \boldsymbol{P}_i \mathbf{x}_{i-1} + (1-\alpha)\mathbf{x}_0$ into

$$\mathbf{x}_i = \alpha((\underline{\boldsymbol{T}}[\mathbf{x}_{i-1}] + \boldsymbol{A})\boldsymbol{D}_{A_i}^{-1})\mathbf{x}_{i-1} + (1-\alpha)\mathbf{x}_0,$$

where $\boldsymbol{D}_{A_i}^{-1}$ is a diagonal matrix with the i^{th} diagonal entry being the inverse of the sum of edge weights connected to node i in $\boldsymbol{A}_i$ (again, we assume a connected graph so these values are all non-zero). Then,

$$\mathbf{x}_i = \alpha \underline{\boldsymbol{T}}[\mathbf{x}_{i-1}]\boldsymbol{D}_{A_i}^{-1}\mathbf{x}_{i-1} + \alpha \boldsymbol{A}\boldsymbol{D}_{A_i}^{-1}\mathbf{x}_{i-1} + (1-\alpha)\mathbf{x}_0.$$

Set $\mathbf{y}_{i-1} = \boldsymbol{D}_{A_i}^{-1}\mathbf{x}_{i-1}$. Then

$$\mathbf{x}_i = \alpha \underline{\boldsymbol{T}}[\mathbf{x}_{i-1}]\mathbf{y}_{i-1} + \alpha \boldsymbol{A}\mathbf{y}_{i-1} + (1-\alpha)\mathbf{x}_0.$$

This leaves us with the relevant computationally expensive pieces to compute being $\underline{\boldsymbol{T}}[\mathbf{x}_{i-1}]\mathbf{y}_{i-1}$, and the entries in $\boldsymbol{D}_{A_i}^{-1}$. Both will actually involve the same type of operation. Note that, using the definition of $\underline{\boldsymbol{T}}[\mathbf{x}]$ we have that the matrix-vector product $\mathbf{z} = \underline{\boldsymbol{T}}[\mathbf{x}]\mathbf{y}$ has $z_i = \sum_j \sum_k \underline{\boldsymbol{T}}(i,j,k)y(j)x(k)$. Consequently, if we have any means of *iterating* over the triangles of a graph, then we can compute $\underline{\boldsymbol{T}}[\mathbf{x}]\mathbf{y}$ for any pair $\mathbf{x}$ and $\mathbf{y}$ in a fashion akin to a sparse-matrix-vector product but in runtime proportional to the number of triangles in the graph.

This directly enables us to compute $\underline{\boldsymbol{T}}[\mathbf{x}_{i-1}]\mathbf{y}_{i-1}$. To compute the entries in $\boldsymbol{D}_{A_i}^{-1}$, note that $\underline{\boldsymbol{T}}[\mathbf{x}]$ is a symmetric matrix because it can be written as a sum of symmetric matrices (since $\underline{\boldsymbol{T}}$ is fully symmetric in all permutations). Thus, the row-sums of $\boldsymbol{A}_i$ are the vertex-degrees we need to build $\boldsymbol{D}_{A_i}^{-1}$. Let $\mathbf{e}$ be the vector of all ones, then these are computed as $\boldsymbol{A}_i\mathbf{e} = \underline{\boldsymbol{T}}[\mathbf{x}_i]\mathbf{e} + \boldsymbol{A}\mathbf{e}$. Since $\boldsymbol{A}$ is not changing, we only need to compute the column sums of $\underline{\boldsymbol{T}}[\mathbf{x}_i]\mathbf{e}$ at each iteration. Again, we can use an implicit tensor-vector-vector product operation to compute the column sums. Thus, all operations involving the tensor $\underline{\boldsymbol{T}}$ are linear in terms of the number of triangles in the network.

Convergence of this type of nonlinear system of equations is theoretically delicate with bounds that are often insufficient for practice [26]. Empirically, we observe that the iterations converge. However, absent a robust theory, this method is only run for a small and fixed number of iterations (10). This will produce a unique deterministic and reproducible set of scores that locally capture the influence of both the graph and the reinforced triangles.

For ease of reuse, we provide our code for TRPR

`https://github.com/nassarhuda/pairseed`.

IV. Experimental Setup

We now perform a series of experiments on synthetic as well as real-world graphs from a variety of disciplines, including online social networks, communication networks, and biological interaction networks. We include experiments for static networks as well as a temporal network and find that in both cases, our pairseeded PageRank and TRPR methods consistently outperform the baseline measures in terms of AUC score, often substantially.

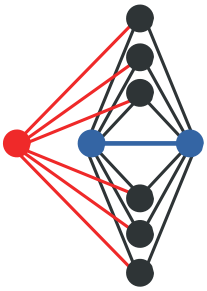

Fig. 2. Motivating social network example for the TRPR algorithm. If all of the *friends* of the blue couple know the red node, we want to predict that the red node must know the blue couple as well. Running TRPR on the above example, with $\mathbf{e}_{u,v}$ being a seed vector on the blue nodes reveals that the red node has the third highest score after the two blue nodes. After 10 iterations of Algorithm 1 with $\alpha = 0.8$, the ouput vector assigns a score of 0.102 to the red node, 0.063 to the black nodes, and 0.257 to the blue nodes.

Algorithm 1: TRPR

Input: $\underline{\boldsymbol{T}}, \boldsymbol{A}, \alpha, \mathbf{e}_{u,v}$, number of iterations n
Output: $\mathbf{x}$
$\mathbf{x}_0 = \mathbf{e}_{u,v}$
for $i = 1, 2, \ldots, n$ **do**
  $\hat{\boldsymbol{X}}^{(i)} = \underline{\boldsymbol{T}}[\mathbf{x}_{i-1}]$ # i.e. $\hat{X}^{(i)}_{r,s} = \sum_k \underline{\boldsymbol{T}}(r,s,k)\mathbf{x}_{i-1}(k)$
  $\boldsymbol{P}_i = \text{normalize}(\hat{\boldsymbol{X}}^{(i)} + \boldsymbol{A})$ # column stochastic
  $\mathbf{x}_i = \alpha \boldsymbol{P}_i \mathbf{x}_{i-1} + (1-\alpha)\mathbf{x}_0$
return $\mathbf{x}_n$

In this section, we lay out the setup of our main types of experiments and summarize the methods that we evaluate. The specific scenarios investigated are discussed in the subsequent sections along with the results.

A. Wedges experiments

The wedges experiments are akin to the leave-p-out cross validation metric, in the sense that we will drop a select number of edges from the network and use them as a validation set. An experimental trial in this setting is designed as follows. For a given graph, we first randomly pick an edge in this graph (call it the seed edge) and find all of the triangles to which it belongs. Next, we drop one edge from each triangle uniformly at random, which transforms each triangle containing the seed edge into a length-2 path, or wedge. For evaluation, we use a pairwise link prediction method on the seed edge, which produces an ordering on the nodes. Given this ordering, we compute the area under the ROC curve (henceforth, *AUC score*) as a measure of performance. The experiment is repeated for a fixed number of edges.

B. 80-20 experiments

The 80-20 experiments are akin to hold-out cross validation and are a standard way of evaluating the classical link prediction problem. In this setup, for a given network, we drop 20% of the edges and label them as testing data, and use the remaing 80% as training data to make predictions. Then, for random edges in the training data (call them seed edges), we use the pairwise link prediction methods to predict which nodes will form triangles with each edge that is selected. A "correct" prediction is a node that forms a triangle with the seed edge when including the test data, but at least one edge from the triangle is missing in the training data. Again, for

a given edge, each method produces a similarity score on all nodes, and we use the ordering of the nodes induced by the scores to determine the AUC score.

We also perform a similar experiment on a temporal network (CollegeMsg) with timestamps on the edge arrivals. In this scenario, the dropped 20% edges are not chosen at random. Instead, we split the data into training and test sets based on the time—the first 80% of the edge to appear in time are the training data and the remaining 20% are the test data.

C. Summary of methods and parameter settings

Finally, we summarize all of the methods that we use for pairwise link prediction.

- Pairseed: This is our method described in Section III-B. We use the implementation from `MatrixNetworks.jl` [27] with $\alpha = 0.8$. This implementation solves the linear system until convergence to machine precision.
- Single seed (SS): For comparison, we present the results of single-node seeded PageRank on an arbitrary end points of the seed edge. We use the same implementation as above, with $\alpha = 0.8$.
- TRPR: This is our method described in Section III-C. We use $\alpha = 0.8$ and number of iterations $n = 10$.
- AA, PA, JS: For a seed edge, we compute the generalized Adamic-Adar, Preferential Attachment, and Jaccard similarity scores, respectively (as in Section III-A) between the seed edge and all remaining nodes in the graph.

V. Pairwise link prediction results

For the results in this section, we report the distribution of AUC scores of our predictions over 300 random experiments.

A. Synthetic experiments

In this set of experiments, we use preferential attachment and stochastic block model networks, generating 600-node graphs as input to the wedges and 80-20 experiments.

Generalized Preferential Attachment (GPA). Generalized Preferential Attachment [28] is a graph generation model that generalizes the classical preferential attachment model [17] that permits the addition of new components at each step of the algorithm, and not just nodes. In our generation of synthetic graphs, the event of node addition occurs with probability $1/2$, and the event of edge addition occurs with probability $1/2$. The starting graph structure is a clique of size 5. At each step of the graph generation process, an edge or node is added by attaching proportionally to the degrees of the existing nodes.

Stochastic Block Model (SBM). In this model, we plant three communities of size 200 nodes each, where the edge probability within each community is $20/200$ and the edge probability between any two communities is $10/200$. When testing our methods on the SBM model, we keep the connections within communities and only hide edges across communities to be predicted. In the *wedges experiment*, we only select edges between two communities and find their corresponding third node to form triangles with from the third community.

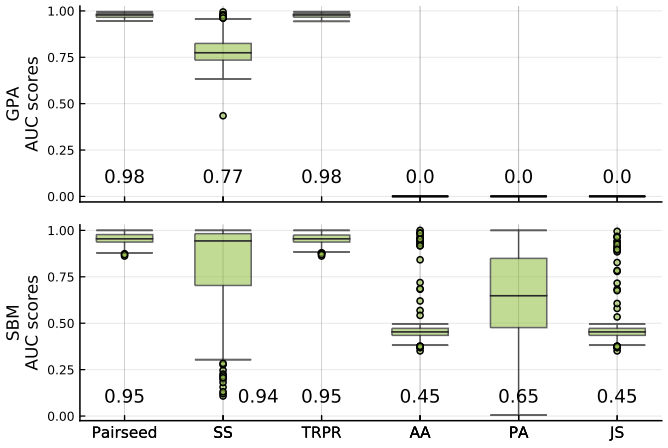

Fig. 3. Box plots of the AUC scores distribution from the wedges experiment applied on synthetic graphs under the GPA (top panel) and SBM (bottom panel) models. The numbers are the median value. Pair-seeded PageRank and TRPR are our proposed methods for pairwise link prediction and they are consistently better than other single-seeded PageRank and the local similarity metrics. See the discussion in the text about AA, PA, and JS on the GPA experiment.

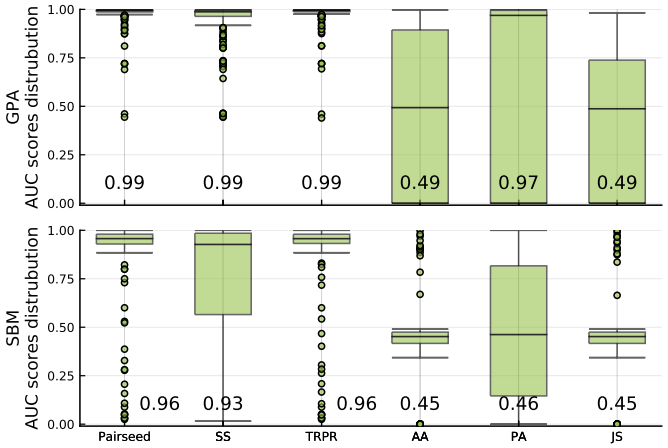

Fig. 4. Box plots of the AUC scores distribution from the 80-20 experiment. The numbers are the median value. In both scenarios, GPA (top panel) and SBM (bottom panel), we observe that pair-seeded PageRank and TRPR are superior to other methods.

Results. Figures 3 and 4 show the results of the pairwise link prediction for the wedges experiments and 80-20 experiments, respectively. In all experiments, our pair-seeded PageRank and TRPR methods produce better results than the single-seeded method and the local similarity measures. For the results in Figure 3 with the GPA model, the setup of the wedges experiment forces the constraint that any edge (u, v) used for prediction participates in no triangles in the training data. Thus, $\Gamma((u, v))$ is empty and the local methods evaluate to zero. In the SBM experiment, we only removed certain types of edges, so the local methods do not always evaluate to zero.

The results in Figure 4 for the PA baseline makes many good predictions on the GPA graph, with the median close to 1. This is not a surprise since the GPA graph model builds the graph by appending new nodes or edges to existing nodes with probability proportional to the degrees of the existing nodes.

TABLE I
STATISTICS OF THE REAL-WORLD DATASETS USED IN THIS PAPER.

Network name	nodes	edges	type
Penn94	41536	1362220	Social
MU78	15425	649441	Social
Caltech36	762	16651	Social
Ch-Ch-Miner	1510	48512	Biology
P-P-Pathways	21521	338624	Biology
email	1133	5451	Comm.
CollegeMsg	1899	13838	Temporal

B. Real-world graphs

In this section we perform our pairwise link prediction experiments on six real-world graphs from different domains. The networks are as follows:

- Penn94, MU78, and Caltech36 are online social networks from the Facebook100 collection of datasets [29].
- Ch-Ch-Miner is a biological network of drug (chemical) interactions [30], [31].
- P-P-Pathways is a biological network of physical interactions between proteins in humans [32].
- email is an email communication network [33].

Table I provides summary statistics for these datasets, along with the temporal network we use in the next section.

Figure 5 shows the results of the wedges experiments. Recall that the baseline measures do not produce meaningful output for this experiment. Again, we see that pair-seeded PageRank and TRPR consistently produce better performance as compared to single-seeded PageRank. The results of the 80-20 experiments for the online social networks are in Figure 6. Here, we observe an improvement of the local methods such as Adamic-Adar and Jaccard Similarity. Nevertheless, this is not unexpected, as these metrics model the social interactions paradigm such as having a large number of common friends. We show the 80-20 experiment on the remaining networks in Figure 7. The one scenario in which a local method outperforms the diffusion methods is the protein-protein interaction network where the PA metric produces the best performance. This might not be surprising, as studies have suggested that preferential attachment governs the evolution of protein-protein interaction networks [34].

C. Case study on a temporal graph

As stated in the introduction, link prediction can be viewed as the task of predicting missing links from the network or future hidden links. With our wedges and 80-20 experiments, we modeled the prediciton task of missing links. Next, we intend to focus on a case study of predicting future links on a communication network. We use the temporal network CollegeMsg [35], which has timestamped edges between students on a college's private messaging service. The summary statistics of the dataset are in Table I.

We run the 80-20 experiment on this network while treating the earliest 80% of the unique edges as our training data and the remainder in the test data. We use two evaluations here:

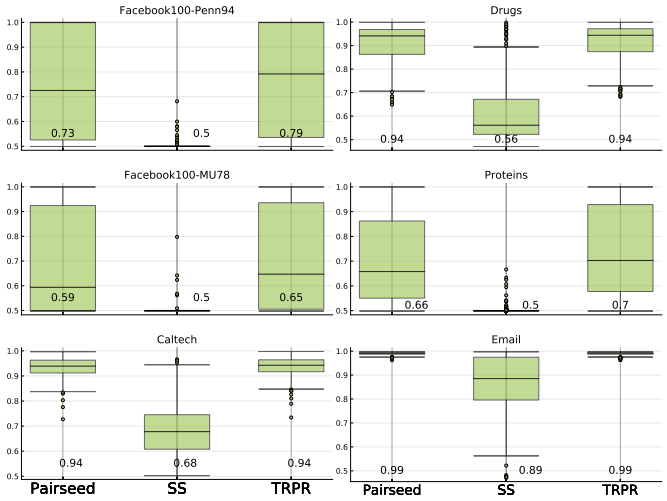

Fig. 5. Results of the wedges experiment on six real-world networks from Table I. Pair-seeded PageRank and TRPR produce superior results as compared to single-seeded PageRank in all experiments.

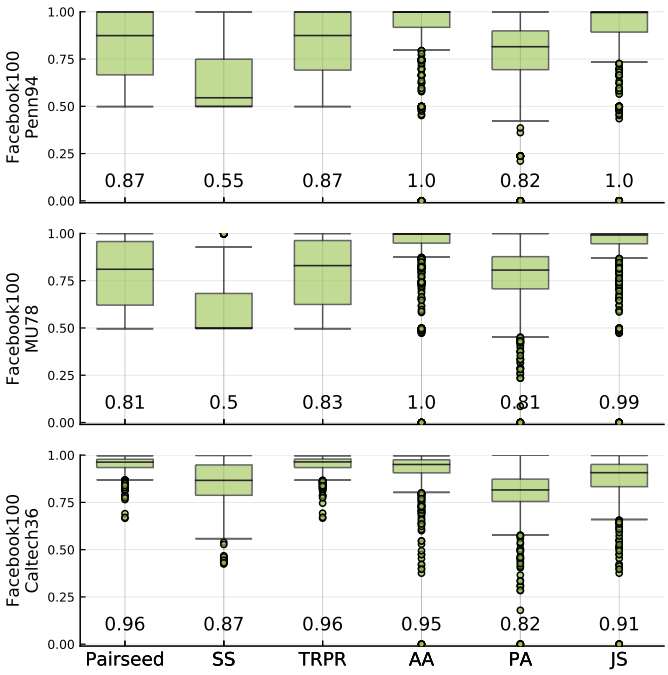

Fig. 6. Results of the 80-20 experiments on the three Facebook100 networks. AA and JS produce the best results on Penn94 and MU78. These local methods give comparable results compared to TRPR and pair-seeded PageRank on the Caltech36 network. Pair-seeded PageRank and TRPR still outperform single-seeded PageRank.

- The "AND" metric (top panel of Figure 8) considers predictions on nodes where both edges connecting the node to the seed edge are in the test data.
- The "OR" metric (bottom panel of Figure 8) considers predictions on nodes where exactly one of the two edges connecting the node to the seed edge is in the test data.

Figure 8 shows that our pair-seeded PageRank and TRPR outperform all the other link prediction metrics, and thus, both pair-seeded PageRank and TRPR are well suited for pairwise link prediciton on temporal networks.

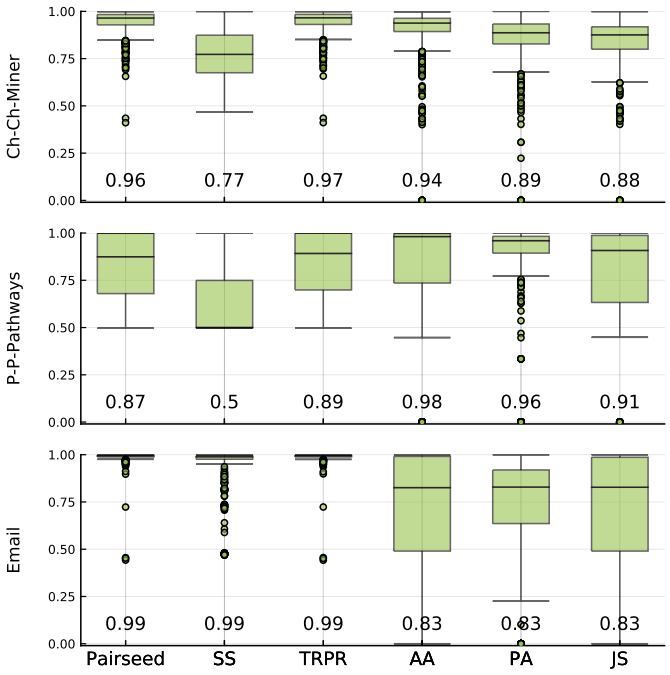

Fig. 7. Results of the 80-20 experiment on the biological and email networks. PA can perform better than diffusion type methods on protein-protein interaction networks; however, this result agrees with the literature showing that such networks evolve preferentially.

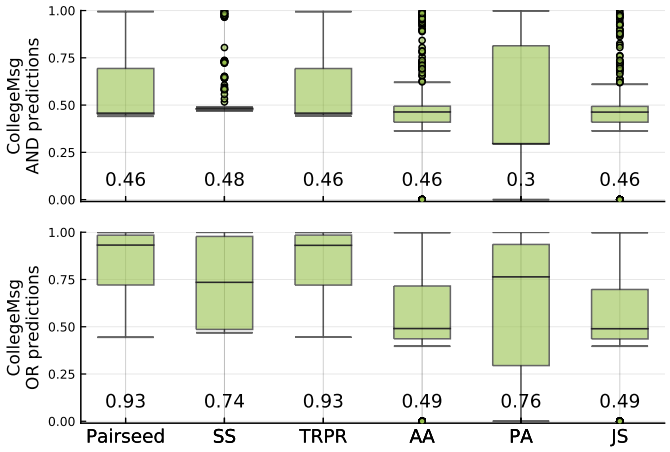

Fig. 8. The 80-20 experiment applied on the CollegeMsg temporal network where training and test data are split by edge arrival time. The "AND" metric (top panel) measures pairwise link prediction performance for triangles where both edges connecting the predicted node to the seed edge are in the test data. The "OR" metric (bottom panel) measures performance where only one of the two triangle edges connecting the predicted node to the seed edge is in the test data. The "AND" metric is more challenging, and performance is generally worse compared to the "OR" metric. Regardless, in both cases, our pair-seeded PageRank and TRPR methods have the best prediction performance.

VI. Back to Standard Link Prediction

In this section we bring our attention back to the standard link prediction problem and show how the methods we presented in this paper can also be used to further enhance standard link prediction. We split our data in the same way to the 80-20 experiments. Then, for the top 100 nodes with the largest degree in the training data, we perform different types of seeded PageRank diffusion for link prediction on these nodes. This choice of nodes serves the purpose of identifying nodes that have a higher chance of making connections in the test data. Again, we measure performance in terms of AUC scores. Our baseline is single-seeded PageRank.

TABLE II
DESCRIPTION OF METHODS INSPIRED BY PAIRWISE LINK TO PERFORM THE STANDARD LINK PREDICTION TASK

sum▲	For a certain node i, aggregate the pair-seeded PageRank results from all edges adjacent to i. This is equivalent to performing PageRank with a normalized initial vector valued 1 at the indices of all the neighbors of i, and degree(i) at index i.
max●	This is similar to the previous approach, but we instead take the element-wise maximum value of the pair-seeded PageRank vectors.
star-seed+	This is similar to pair-seeded PageRank, except that we start PageRank with a normalized initial vector valued 1 at the index of the seed node and all its neighbors.
TRPR♦	This uses the same starting vector as star-seed, but instead, applies the TRPR algorithm on it.

Our results on pairwise link prediction suggest that multiple seeds with PageRank-like methods are effective. Here, we consider four multiple-seeding strategies and compare them to single-seeded PageRank for the classical link prediction problem. We summarize the four new methods in Table II.

We use real-world networks from Section V-B, and present our results in Figure 9. The scatter plots compare the AUC score of the neighborhood-based seeding methods to the AUC scores from single-seeded PageRank. These results suggest that neighborhood-based seeding are superior to single-seeded PageRank as a link prediction method.

VII. Discussion and Future Work

Having a reliable link prediction algorithm is an well-studied research topic due to its utility in many disciplines. Traditional link prediction methods aim to find pairs of nodes that are likely to form a link. Here, we have studied a higher-order version of the problem called pairwise link prediction where we predict nodes that are likely to form a triangle with an edge. We generalized local link-prediction methods and we developed two PageRank-based methods for this problem. These PageRank-based methods generally out-performed extensions of local link prediction methods on a variety of datasets. Using these results as inspiration, we then developed multiple-seeding strategies for PageRank in classical link prediction, which outperform their standard single-seeded counterparts.

While we haven't focused on computational efficiency for the sake of space, we note that highly efficient implementations of our procedures are possible given their close relationships with traditional PageRank methods. Scaling to billions

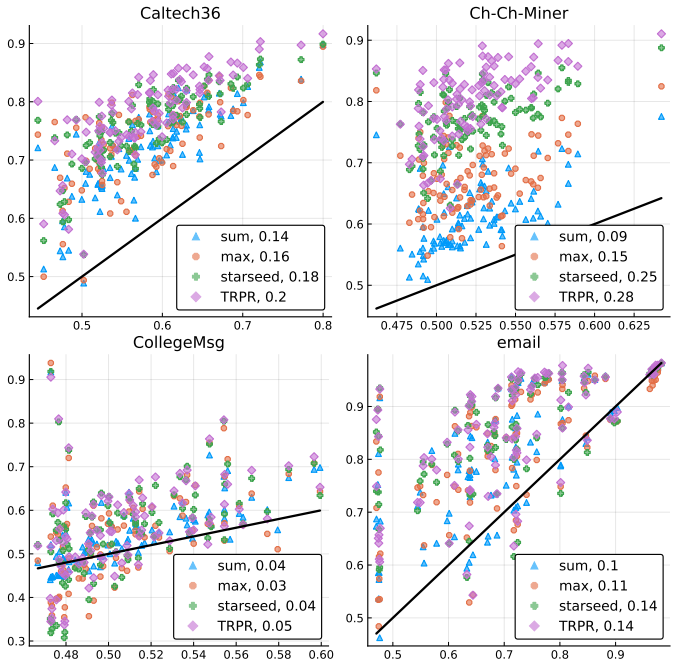

Fig. 9. Results of standard link prediction experiment on four real-world networks. Each scatter plot shows the link prediction AUC results of 100 experiments of methods inspired by our pairwise link prediction proposal with respect to the AUC scores of single-seeded PageRank. The solid black line is the plot of $f(x) = x$. Points above the line are cases where our proposed methods have superior performance to standard single-seeded PageRank. We see that in most cases the four methods outperform the classical seeded PageRank method. This study suggests that it is useful to consider a node's neighborhood for the purposes of seeding for link prediction with PageRank. The values in the legend serve as a summary performance measure, which is the average distance to the $f(x) = x$ line.

of nodes and edges is simply not a problem given current abilities to compute PageRank [25]. The space of higher-order prediction problems also has limitless sub-structure. An alternate problem is to predict an edge that is important when given a single node. In the future, we intend to extend this work to the latter scenario, and TRPR can be adapted for this purpose.

References

[1] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 7, 2007.

[2] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 6, 2011.

[3] A. Clauset, C. Moore, and M. E. J. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, pp. 98 EP –, May 2008. [Online]. Available: https://doi.org/10.1038/nature06830

[4] L. Backstrom and J. Leskovec, "Supervised random walks: Predicting and recommending links in social networks," ser. WSDM '11. New York, NY, USA: ACM, 2011, pp. 635–644.

[5] C. A. Gomez-Uribe and N. Hunt, "The netflix recommender system: Algorithms, business value, and innovation," *ACM Trans. Manage. Inf. Syst.*, vol. 6, no. 4, pp. 13:1–13:19, Dec. 2015.

[6] C.-H. Lin, D. M. Konecki, M. Liu, S. J. Wilson, H. Nassar, A. D. Wilkins, D. F. Gleich, and O. Lichtarge, "Multimodal network diffusion predicts future diseasegenechemical associations," 10 2018.

[7] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, "Network motifs: Simple building blocks of complex networks," *Science*, vol. 298, no. 5594, pp. 824–827, 2002. [Online]. Available: https://science.sciencemag.org/content/298/5594/824.full.pdf

[8] R. Milo, "Superfamilies of evolved and designed networks," *Science*, vol. 303, no. 5663, pp. 1538–1542, mar 2004.

[9] A. R. Benson, D. F. Gleich, and J. Leskovec, "Higher-order organization of complex networks," *Science*, vol. 353, no. 6295, pp. 163–166, Jul. 2016.

[10] A. R. Benson, R. Abebe, M. T. Schaub, A. Jadbabaie, and J. Kleinberg, "Simplicial closure and higher-order link prediction," *Proceedings of the National Academy of Sciences*, nov 2018.

[11] R. Lambiotte, M. Rosvall, and I. Scholtes, "From networks to optimal higher-order models of complex systems," *Nature Physics*, vol. 15, no. 4, pp. 313–320, Mar. 2019.

[12] D. Easley, J. Kleinberg *et al.*, *Networks, crowds, and markets*. Cambridge university press Cambridge, 2010, vol. 8.

[13] P. W. Holland and S. Leinhardt, "A method for detecting structure in sociometric data," in *Social Networks*. Elsevier, 1977, pp. 411–432.

[14] M. S. Granovetter, "The strength of weak ties," in *Social Networks*. Elsevier, 1977, pp. 347–367.

[15] A. Rapoport, "Spread of information through a population with socio-structural bias: I. assumption of transitivity," *The Bulletin of Mathematical Biophysics*, vol. 15, no. 4, pp. 523–533, Dec. 1953.

[16] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Social Networks*, vol. 25, no. 3, pp. 211 – 230, 2003.

[17] M. E. J. Newman, "Clustering and preferential attachment in growing networks," *Phys. Rev. E*, vol. 64, p. 025102, Jul 2001.

[18] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999. [Online]. Available: https://science.sciencemag.org/content/286/5439/509.full.pdf

[19] L. Katz, "A new status index derived from sociometric analysis," *Psychometrika*, vol. 18, no. 1, pp. 39–43, Mar 1953.

[20] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." Stanford InfoLab, Technical Report 1999-66, November 1999, previous number = SIDL-WP-1999-0120.

[21] N. Eikmeier, A. S. Ramani, and D. F. Gleich, "The hyperkron graph model for higher-order features," in *IEEE International Conference on Data Mining, ICDM 2018, Singapore, November 17-20, 2018*, 2018.

[22] H. Nassar, A. R. Benson, and D. F. Gleich, "Pairwise link prediction," *arXiv*, 2019. [Online]. Available: https://arxiv.org/pdf/1907.04503.pdf

[23] R. Andersen, F. Chung, and K. Lang, "Local graph partitioning using PageRank vectors," in *2006 47th Annual IEEE Symposium on Foundations of Computer Science*. IEEE, 2006.

[24] D. F. Gleich, "PageRank beyond the web," *SIAM Review*, vol. 57, no. 3, pp. 321–363, Jan. 2015.

[25] P. Lofgren, S. Banerjee, and A. Goel, "Personalized pagerank estimation and search: A bidirectional approach," in *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, ser. WSDM '16. New York, NY, USA: ACM, 2016, pp. 163–172.

[26] A. Benson, D. F. Gleich, and L.-H. Lim, "The spacey random walk: a stochastic process for higher-order data," *SIAM Review*, vol. 59, no. 2, pp. 321–345, May 2017.

[27] H. Nassar and D. Gleich, "Matrixnetworks.jl," https://github.com/nassarhuda/MatrixNetworks.jl, Oct. 2018.

[28] C. Avin *et al.*, "Core size and densification in preferential attachment networks," in *International Colloquium on Automata, Languages, and Programming*, ser. ICALP 2015. Berlin, Heidelberg: Springer-Verlag, 2015, pp. 492–503.

[29] A. L. Traud, P. J. Mucha, and M. A. Porter, "Social structure of facebook networks," *CoRR*, vol. abs/1102.2166, 2011.

[30] D. S. Wishart *et al.*, "DrugBank 5.0: a major update to the DrugBank database for 2018," *Nucleic Acids Research*, Nov. 2017.

[31] Stanford SNAP Group, "Miner: Gigascale multimodal biological network," https://github.com/snap-stanford/miner-data, 2017.

[32] M. Agrawal, M. Zitnik, and J. Leskovec, "Large-scale analysis of disease pathways in the human interactome," in *Pacific Symposium on Biocomputing*, vol. 23. World Scientific, 2018, p. 111.

[33] R. Guimerà, L. Danon, A. Daz-Guilera, F. Giralt, and A. Arenas, "Self-similar community structure in a network of human interactions," *Phys. Rev. E*, vol. 68, p. 065103, Dec 2003.

[34] E. Eisenberg and E. Y. Levanon, "Preferential attachment in the protein network evolution," *Phys. Rev. Lett.*, vol. 91, p. 138701, Sep 2003.

[35] P. Panzarasa, T. Opsahl, and K. M. Carley, "Patterns and dynamics of users' behavior and interaction: Network analysis of an online community," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 5, pp. 911–932, May 2009.