A Transformative Topological Representation for Link Modeling, Prediction and Cross-Domain Network Analysis

Kai Zhang, Junchen Shen, Gaoqi He, *Member, IEEE*, Yu Sun, Haibin Ling, Hongyuan Zha, Honglin Li, and Jie Zhang

***Abstract*—Many complex social, biological, or physical systems are characterized as networks, and recovering the missing links of a network could shed important lights on its structure and dynamics. A good topological representation is crucial to accurate link modeling and prediction, yet how to account for the kaleidoscopic changes in link formation patterns remains a challenge, especially for analysis in cross-domain studies. We propose a new link representation scheme by projecting the local environment of a link into a "dipole plane", where neighboring nodes of the link are positioned via their relative proximity to the two anchors of the link, like a dipole. By doing this, complex and discrete topology arising from link formation is turned to differentiable point-cloud distribution, opening up new possibilities for topological feature-engineering with desired expressiveness, interpretability and generalization. Our approach has comparable or even superior results against state-of-the-art GNNs, meanwhile with a model up to hundreds of times smaller and running much faster. Furthermore, it provides a universal platform to systematically profile, study, and compare link-patterns from miscellaneous real-world networks. This allows building a global link-pattern atlas, based on which we have uncovered interesting common patterns of link formation, i.e., the bridge-style, the radiation-style, and the community-style across a wide collection of networks with highly different nature.**

***Index Terms*—Complex networks, graph neural networks, link prediction, topological representation.**

Manuscript received 14 June 2023; revised 16 January 2024; accepted 10 March 2024. Date of publication 19 March 2024; date of current version 6 August 2024. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFC3400501, and in part by the National Natural Science Foundation of China under Grant 62276099. The work of Honglin Li was supported in part by the National Science Fund for Distinguished Young Scholars of China under Grant 81825020. Recommended for acceptance by P. Mordohai. *(Corresponding authors: Jie Zhang; Honglin Li.)*

Kai Zhang, Junchen Shen, and Gaoqi He are with the School of Computer Science and Technology, East China Normal University, Shanghai 200062, China (e-mail: kzhang@cs.ecnu.edu.cn; 51215901048@stu.ecnu.edu.cn; gqhe@cs.ecnu.edu.cn).

Yu Sun is with Indeed Inc., Sunnyvale, CA 94086 USA (e-mail: sunyu9910@gmail.com).

Haibin Ling is with the Department of Computer Science, Stony Brook University, Stony Brook, NY 11794 USA (e-mail: hling@cs.stonybrook.edu).

Hongyuan Zha is with the School of Data Science, Chinese University of Hong Kong, Shenzhen, Guangdong 518172, China (e-mail: zhahy@cuhk.edu.cn).

Honglin Li is with the Innovation Center for AI and Drug Discovery, East China Normal University, Shanghai 200062, China, and also with Lingang Laboratory, Shanghai 200031, China (e-mail: hlli@hsc.ecnu.edu.cn).

Jie Zhang is with the Institute of Science and Technology for Brain-inspired Intelligence, Fudan University, Shanghai 200438, China (e-mail: jzhang080@gmail.com).

This article has supplementary downloadable material available at https://doi.org/10.1109/TPAMI.2024.3378729, provided by the authors.

Digital Object Identifier 10.1109/TPAMI.2024.3378729

I. Introduction

MANY complex social, biological, or physical systems are characterized as networks, where vertices represent individual agents and links signify their interactions [1], [2], [3], [4], [5], [6], [7], [8], [9]. Due to the cost and uncertainties of data acquisition, networked data are often incomplete with missing links. As a result, estimating the likelihood that an unobserved link actually exists based on the observed portion of the network, commonly known as link prediction, thus becomes a fundamental problem in network and information sciences [10], [11], [12], [13], [14], [15], [16]. Accurate link prediction is not only a practical goal in physics, social networks and recommender systems, but also provides valuable insights into scientific discoveries related to network structure, dynamics, and organizing principles.

Link prediction is a statistically hard problem and no single model has shown to be superior for all networks unless by stacking different models together as an ensemble model [16]. The predictability is related to the intrinsic structural regularity of the network [15]. Early work study rules of link formation and design various heuristics to evaluate the proximity between two nodes for link prediction [17], [18], [19], [20], [21]. Probabilistic models [22], [23], [24] and maximum likelihood approaches [11], [12], on the other hand, estimate the probability of a link conditioned on the network structure or node attributes. Link prediction can also be solved by using latent node representations through graph embedding techniques [25], [26], [27], [28], [29].

In recent years, learning-based algorithms that predict the missing links through a classifier [30] began to draw more attention. The key advantage of learning based algorithms nowadays lies in their ability to automatically craft features for the predictive task through end-to-end optimization [31] by leveraging a specific inductive bias. In fact, the grand success of deep neural networks is largely attributed to their power of learning good representations. This philosophy has inspired a surge of interest in applying graph neural networks (GNNs) [27], [29], [32], [33], [34], [35] to extract useful topological features for link prediction. The pioneering idea of GNN-based link prediction is due to Zhang et al. [36], which extracts the "*local enclosing subgraph*" for a target link to capture the key topological information for a link to be formed. By doing this, link prediction is converted

to subgraph classification, on top of which powerful GNNs can be readily introduced to generate significantly improved results over previous methodologies.

Despite recent progresses, substantial challenges persist in link representation learning. This is because the links of a network are often enclosed in local subgraphs involving an arbitrary number of nodes and kaleidoscopic topological variation, which is notoriously hard to align or profile. Therefore GNN models may have to use heuristics such as truncation or padding to obtain constant-sized features [37], which inevitably alters graph topology; the convenient choice of graph pooling, which collapses all the nodes into one, may incur information loss and become the bottleneck as noted by [38], [39]. Moreover, graph convolutions can be difficult to interpret when using topological features for message passing. As to graph embedding techniques [27], [29], [34], the low-dimensional node embedding vectors are defined only for each individual network separately, which limits their utility in cross-domain studies involving multiple networks. Overall, the lack of a good link representation hampers not only link prediction accuracy, but also knowledge discovery from links of miscellaneous real-world networks.

In this paper, we propose a transformative link representation that characterizes complex link patterns in the network with desired interpretability, generalization, and cross-domain modeling capacity. The key idea is to project the local environment (or enclosing-subgraph) of a link into a two-dimensional "*dipole plane*". In this plane, the neighboring nodes of the link are positioned via their relative proximity to the two anchors of the link through random-walks, like a dipole. By doing this, complex and discrete topology arising from link formation is turned to continuous and differentiable distribution of a point-cloud, opening up new possibilities for topological feature-engineering in link modeling and prediction.

The presented link representation offers significant advantages for link modeling and prediction. Theoretically, it possesses the ability to discern between link patterns with different topological characteristics, which is critical for training discriminative models. The density-based profile is also robust against link perturbations due to inherent kernel smoothing, thus greatly contributing to the generalization performance. Empirically, the proposed representation yields comparable or even superior results in link prediction compared to the best-performing Graph Neural Networks (GNNs), yet with a compact model up to hundreds of times smaller and running much faster. Finally, the proposed link representation is physically interpretable and with naturally aligned dimensions, therefore it provides a universal platform to study, compare and explore link-patterns from networks across different domains simultaneously. This allows building a global link-pattern-atlas, based on which we have uncovered interesting general themes of link formation, as well as network similarities that may otherwise be hidden from a collection of scientific, social, biological and technological networks of highly different nature.

The subsequent sections of this paper are structured as follows. Section II provides a review of related work. Section III introduces the proposed link representation scheme, while Section IV contains the theoretical analysis. Experimental evaluations and cross-domain case studies are presented in Section V. The concluding remarks and identification of future directions are outlined in the last section.

II. Related Work

A large body of link prediction methods has been devised and can be broadly grouped into similarity-based approaches, probabilistic models, embedding techniques, and learning-based models, as outlined in relevant surveys [40], [41].

The early efforts of link prediction focus on the design of various heuristics to quantify the structural relations between two nodes. Well-known examples include the common neighbors [17], Jaccard similarity [18] and Adamic–Adar index [19] based on first-order neighbors. Later, high-order heuristics are proposed to characterize the relation between the nodes in a more global context. For example, Rooted PageRank [20] measures the probability of reaching a node from a predefined root node through random-walks, implicitly taking into account all possible paths in the network. Katz score [21] considers paths of varying lengths and assigns higher weights to shorter paths. SimRank [42] evaluates node similarity by recursively looking into their neighbors. These heuristics are easy to compute, and can well recover the missing links if the underlying assumption of homophily is satisfied.

Another big family of algorithms relies on latent node embeddings as fundamental features for link prediction. Such low-dimensional representations can be obtained through traditional matrix factorization technique [25] and stochastic block models [26]. In recent years, the development of distributed representation learning frameworks such as the skip-gram model in word2vec [43] lays a solid foundation for network embedding (or node embedding), with prominent examples including DeepWalk [27], LINE [28] and node2vec [29], and a unified matrix factorization view can be found in [34]. The high-quality network embedding allows faithfully capturing the proximity relation between the nodes of a network, and hence sophisticated classifiers can be built upon pairs of such node embeddings to perform link prediction. See a comprehensive review for such methods in [40].

The concept of using GNN models for link prediction is pioneered by Zhang et al. in their seminal work of "SEAL" [36], which introduces the notion of "*local enclosing subgraphs*". Given a graph G and any pair of nodes x and y representing a target link, the local enclosing subgraph G_{xy} is defined as a subgraph that is composed of the union of x and y's neighbors up to l hops. The enclosing subgraph delineates the "l-hop surrounding environment" centered around the target link (x, y), encapsulating the crucial topological information necessary for link formation by theoretically approximating a wide range of high-order heuristics. It allows link prediction to be converted faithfully to subgraph classification. Automatic graph feature engineering can then be performed on top of the enclosing subgraphs with GNNs, leading to promising results that significantly outperform previous algorithms.

Inspired by SEAL [36], various innovations have been made toward extracting features from the enclosing subgraph (or the

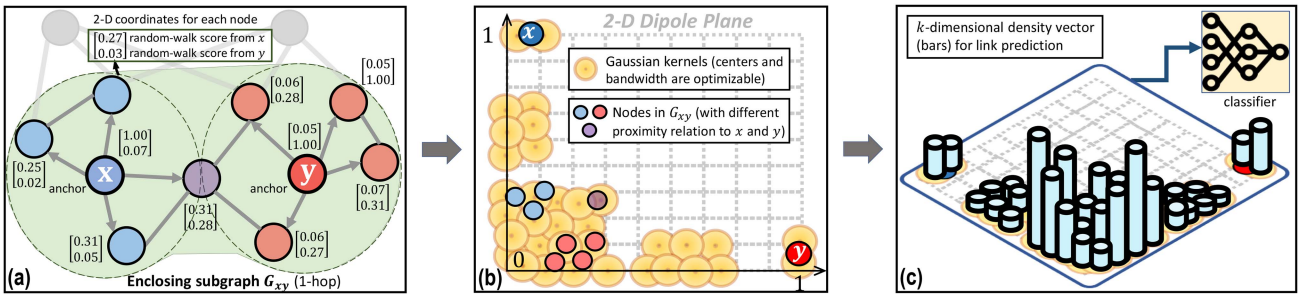


Fig. 1. Proposed Dipole Space Density Network (DSDN) that transforms a link to a continuous point-cloud distribution through an end-to-end learning architecture. (a) For a focal link between two anchor nodes x and y, find the l-hop (here $l = 1$) enclosing-subgraph G_{xy}. Then use two random-walks starting from x and y to measure the proximity between each node in G_{xy} and the two anchors. (b) Project the nodes in G_{xy} to the "dipole plane", where the 2-D coordinates are exactly the scores generated by the two random-walks starting from x and y. Then estimate the density of the 2-D point-cloud using k Gaussian kernels whose centers and bandwidth are both optimizable. (c) Use the estimated density as input features to a classifier for link prediction.

entire training graph). For example, Walk-Pooling [39] integrates node representations and graph topology into random-walk based transition probabilities, and uses the difference of such probabilities before and after removing the focal link of the enclosing subgraph as the high-order structural information for link prediction. Line-Graph Network [38] proposes to convert the enclosing subgraph to a line graph where each node corresponds to a unique link in the original graph. The features of the link can then be learned directly through the line graph representation, turning link prediction problems to node classification with significantly improved performance. Distance-enhanced GNN [44] combines the pairwise node distances with the GNN model, and obtains promising results in the problems of drug-drug-interaction and protein-protein-association. The distance calculations are based on a set of random anchors to improve the computational efficiency.

III. A New Representation for Link Prediction

Let $G_0 = (\mathbb{V}_0, \mathbb{E}_0)$ be a "complete" network with edge set $\mathbb{E}_0$ and node set $\mathbb{V}_0$. In practice, only a partial version $G = (\mathbb{V}, \mathbb{E})$ is observed such that $\mathbb{E} \in \mathbb{E}_0$, $\mathbb{V} \in \mathbb{V}_0$. The goal of link prediction is then to predict whether a link indeed exists between a pair of unconnected nodes, $(x, y) \in \{\mathbb{V} \times \mathbb{V} - \mathbb{E}\}$, based on the observed graph G.

A focal link between any two nodes, x and y, is characterized by its *enclosing subgraph* G_{xy} as in Fig. 1(a), i.e., a subgraph composed of the l-hop neighbors around x and y. In constructing the enclosing subgraph, l is typically chosen as a small integer like one or two, and the truly observed focal link between x and y will be removed from the subgraph. Next, we show how to obtain a desired representation of the link between x and y based on its enclosing subgraph.

The proposed approach is called "Dipole Space Density Network (DSDN)" and illustrated in Fig. 1. First, we use two random-walks that start from the two anchors of the link, x and y, to evaluate the proximity between the neighboring nodes and the two anchors. Then the "local environment" of the target link G_{xy} can be projected as a point-cloud onto a 2-D "dipole plane", wherein the distribution closely mirrors the local topological organization of the target link. Finally, an adaptive kernel estimator is used to profile the density distribution as a compact and informative representation for link prediction. In the following, we introduce method details.

A. From Enclosing-Subgraph to Point-Cloud

We first use two random-walks starting from the two anchors of the target link, x and y, to evaluate the proximity between the nodes in G_{xy} and the two anchors. More specifically, we resort to Random-walk With Restart (RWR) to quantify such relation [45]. It starts from one anchor node x and iteratively visits neighbors encountered, each step having a probability $1 - c$ to jump back to the start node x, as

$$\mathbf{p}_x^{(t+1)} = c \cdot \tilde{A}_{xy}\mathbf{p}_x^{(t)} + (1 - c) \cdot \mathbf{e}_x. \tag{1}$$

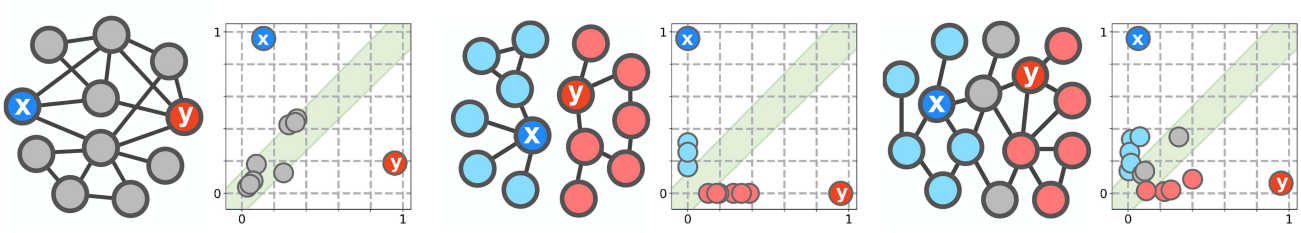
Here t is step, $\mathbf{p}_x \in \mathbb{R}^{n_{xy} \times 1}$ is node-wise random-walk scores, n_{xy} is the number of nodes in G_{xy}, $\tilde{A}_{xy}$ is the transition probability matrix of G_{xy}, $1 - c$ is the restart probability, and $\mathbf{e}_x$ is a one-hot vector indicating the start node x. The converged distribution for $t \to \infty$ has closed form [45],

$$\mathbf{p}_x = (\mathbf{I} - c \cdot \tilde{A}_{xy})^{-1}\mathbf{e}_x. \tag{2}$$

The random-walk score $\mathbf{p}_x$ reflects the proximity of each node to anchor x. If $c = 0$, the re-start probability $1 - c$ is 1, namely the random-walk will be frozen at x with $\mathbf{p}_x$ being a one-hot vector; if $c \to 1$, $\mathbf{p}_x$ becomes the stationary distribution of a Markov random-walk that freely explores G_{xy} but never jumps back to x; in other words, the start node is forgotten. In practice, we use $c \in [0.5, 0.8]$ to explore the subgraph G_{xy} thoroughly while still remembering where the starting node is. We also prefer the closed-form random-walk distribution (2) because it is more robust than the discrete version (1) and saves the effort of determining the number of iterations.

We use two separate random-walks from x and y as in Fig. 1(a), to map G_{xy}'s nodes to a new space with n_{xy} pairs of coordinates $\mathbf{P}_{xy} = [\mathbf{p}_x\ \mathbf{p}_y] \in \mathbb{R}^{n_{xy} \times 2}$, and normalize them by the maximum score, as

$$\mathbf{P}_{xy} = \left[\frac{\mathbf{p}_x}{\max(\mathbf{p}_x)}\ \ \frac{\mathbf{p}_y}{\max(\mathbf{p}_y)}\right]. \tag{3}$$



(a) An observed link between x and y (2-hop). (b) A NULL link between x and y (2-hop). (c) An observed link between x and y (2-hop).

Fig. 2. Three link examples, each with their enclosing-subgraphs G_{xy} on the left, and resultant point-clouds on the right. The focal link between x and y will be removed if it is truly observed. The 2-D coordinates of the point-cloud are proximities of each node to x and y. Here, blue and red mark the nodes closer to anchor x and y, respectively, while gray nodes along the diagonal are about equally distant away from x and y. (a) nodes in G_{xy} are roughly equal-distant away from x and y, so the point-cloud is distributed along $y = x$; (b) nodes in G_{xy} breaks into two components, so the point-cloud forms two arms along the two axes of the dipole plane; (c) nodes in G_{xy} fall into three zones: closer to anchor x (blue), closer to anchor y (red), or in the middle (gray).

By normalizing the random-walk score vectors $\mathbf{p}_x$ and $\mathbf{p}_y$ with their maximum entries, the resultant point-clouds will always reside in a unit square $[0, 1] \times [0, 1]$. We call it a "*dipole plane*" as in Fig. 1(b), because it is induced by the two anchors of the target link, like a dipole. It serves a universal "coordinate system" quantifying how the neighboring nodes of a link gather around its two anchors.[1]

B. From Point-Cloud to Density-Profile

The "dipole plane" allows any link of interest to be projected and turned to a 2-D point-cloud. *How is subgraph topology translated to point-cloud distribution*? Since the 2-D point coordinates quantify the proximity of a node to the two anchors of the link, x and y, some simple observations can be made: (i) nodes far away from x and y will have lower proximity scores and hence be mapped around the origin, otherwise they will be pushed away from it; (ii) nodes equally distant away from x and y will be mapped around $y = x$. In fact, the symmetry, grouping, and shape of the point-cloud are closely related to the topological characteristics of the link.

Fig. 2 illustrates various scenarios: (a) is an observed link, where the majority of nodes in G_{xy} are approximately equidistant from x and y, resulting in the formation of a diagonal band pattern in the point-cloud; (b) is a NULL link, for which G_{xy} breaks into two components, and so the point-cloud has two arms along the two axes; (c) is another observed link, where blue and red nodes are close to x and y, respectively, and gray nodes occupy the middle, resulting in a mixed distribution pattern in the point-cloud. In all cases, anchor x and y are around the corners. Sometimes the points aggregate into clusters in the dipole plane, meaning that nodes within a cluster share similar proximity patterns to x and y.

Next we show how to model these highly diversified point-cloud distributions such that a compact, informative representation can be obtained for link prediction. A crucial observation is that not all locations in the dipole plane hold significant density values; some locations could be rarely populated by the point-cloud, while some locations may always have similar density values for true links and NULL ones. Hence, our focus is to identify the locations in the dipole plane whose densities are valuable for distinguishing true links from NULL ones.

To achieve this, we borrow the idea from Parzen window density estimator [46], and extend it to a discriminative version. We place a number of k Gaussian kernels in the dipole plane that are marked as yellow circles in Fig. 1(b), each with center $\boldsymbol{\mu}_j \in \mathbb{R}^{1\times 2}$'s for $j = 1, 2, \ldots, k$, and a common bandwidth $h \in \mathbb{R}_+$. These kernel parameters are then optimized in an end-to-end fashion by minimizing the link-prediction error (defined in Section III.C), so that a faithful density landscape can be recovered for link prediction tasks.

Specifically, we compute the affinity between the n_{xy} points in (3) and the k kernel centers $\boldsymbol{\mu}_j$'s as a $\mathbb{R}^{n_{xy}\times k}$ matrix

$$\mathbf{W}_{ij} = \exp\left(-\frac{\left\|\mathbf{P}_{[i,:]} - \boldsymbol{\mu}_j\right\|^2}{2\,h^2}\right), \tag{4}$$

where $\mathbf{P}_{[i,:]}$ is the i-th row (point) in $\mathbf{P}_{xy}$ (the sub-index xy is omitted for convenience). Namely, if the i-th point $\mathbf{P}_{[i,:]}$ falls in the receptive domain of the j-th Gaussian kernel, it activates the kernel with strength $\mathbf{W}_{ij}$ (like triggering a "sensor"). After normalizing each row of $\mathbf{W}$ to probabilities that sum up to 1, so that its ij-th entry then signifies the probability that the i-th point triggers the j-th kernel, we then sum up all the rows of $\mathbf{W}$ to get the accumulated density at each kernel center $\boldsymbol{\mu}_j$

$$f(\boldsymbol{\mu}_j) = \sum_{i=1}^{n_{xy}} \frac{\mathbf{W}_{ij}}{\sum_{j'=1}^{k} \mathbf{W}_{ij'}}. \tag{5}$$

The point-cloud distribution can then be nicely encoded as the following, k-dimensional density-profile

$$\mathbf{F}_{xy} = \left[f(\boldsymbol{\mu}_1), \quad f(\boldsymbol{\mu}_2), \quad \ldots, \quad f(\boldsymbol{\mu}_k)\right]. \tag{6}$$

The representation $\mathbf{F}_{xy}$ has several advantages: (i) it is easy to compute and can be very compact in size; (ii) it is invariant to the order of the nodes in subgraph G_{xy}; (iii) it has k dimensions that are consistently defined across different subgraphs of a network, even when they have varying number of nodes. In fact, by sharing the same set of Gaussian centers $\{\boldsymbol{\mu}_j's\}$ in the dipole plane, $\mathbf{F}_{xy}$ can be used as an intrinsic, well-aligned link feature even across different networks for cross-domain studies; (iv) it is interpretable. The k dimensions of $\mathbf{F}_{xy}$ sum up to n_{xy}, the

[1]For each link, the two anchors can be either labeled as (x, y) or (y, x), so we can generate two point-clouds that are symmetric with regard to the diagonal of the dipole plane, with the same label.

number of nodes in G_{xy} due to normalization in (5).[2] So the j-th dimension $f(\boldsymbol{\mu}_j)$ can be deemed as the "amount" of nodes in G_{xy} whose relative proximity to the two anchors are close to $\boldsymbol{\mu}_j = [\mu_{j1}\ \mu_{j2}]$, with $\boldsymbol{\mu}_j$ the proximity pattern specified by the j-th Gaussian center. Given this inherent physical interpretation, our network model is no longer a black-box, but instead allows human interpretation of prediction results by looking into dominantly weighted Gaussian kernels in the classifier, each of which represents a specific proximity or topological pattern towards link formation in the enclosing subgraph.

C. Link Prediction

The k-dimensional density profile $\mathbf{F}_{xy}$ in (6) is used as features for a multilayer perceptron (MLP) for link prediction, with the cross entropy loss function. This then leads to a complete end-to-end architecture, in which the Gaussian kernels are placed adaptively to generate compact, discriminative representation. Empirically, $k \in [100, 200]$ suffices for most datasets. This gives a parsimonious model that is not only computationally efficient, but also less prone to overfitting.

In case there are extra node attributes available (such as bags of words feature for a paper in citation networks), the dipole plane provides a platform for heterogeneous message passing, i.e., the cross-talk between the points (nodes) and the "Gaussian sensors" (kernels). We can use the pairwise relation between the n_{xy} points and the k sensors $\mathbf{W}$ as an "adjacency" matrix to aggregate node attributes to the nearby sensors, as

$$\overline{\mathbf{Z}}_{xy} \leftarrow \sigma\left(\mathbf{D}^{-1}\mathbf{W}^{\top}\mathbf{Z}_{xy} \cdot \mathbf{T}\right).$$

Here, $\mathbf{D}$ is column-wise degree matrix of $\mathbf{W}$, $\mathbf{Z}_{xy} \in \mathbb{R}^{n_{xy} \times d}$ is attributes matrix for nodes in subgraph G_{xy}, $\mathbf{T} \in \mathbb{R}^{d \times d'}$ is a linear transform, and $\sigma(\cdot)$ is nonlinear activation. The resultant sensor-wise feature matrix $\overline{\mathbf{Z}}_{xy} \in \mathbb{R}^{k \times d'}$ is of a fixed dimension regardless of the size of the enclosing subgraph G_{xy}, and carries both featural and topological information of the enclosing subgraph. We will feed it into the FC-layers for the final link prediction task.

IV. Properties of the Proposed Link Representation

In this section we study theoretic properties of the proposed link representation. It transforms a target link by two steps, i.e., its enclosing subgraph is first transformed to a 2-D point-cloud, and then to a fixed-dimensional density profile. Interestingly, these two steps exhibit complementary properties. On the one hand, transforming an enclosing subgraph to a point-cloud is a mapping that effectively captures topological differences of enclosing subgraphs. On top of such sensitivity, the kernel density estimator further improves the stability of the learned feature, which is beneficial to link prediction.

[2]In this sense, $\mathbf{F}_{xy}$ is not strictly a density because it sums up to n_{xy}, the subgraph size. We believe n_{xy} carries useful information in link prediction so we do not normalize $\mathbf{F}_{xy}$ to sum to 1.

A. Discriminative Properties of the Point-Cloud

A natural question on the representation power of the point-cloud is under what condition it can distinguish between two different enclosing subgraphs, so that accurate link prediction can be made based on the topological structure of subgraphs.

For simplicity, we consider one anchor x in each subgraph with point-cloud coordinate p_x as in (2). This is because the choice of the two anchors is independent and so the extension from one anchor to two anchors is trivial. Second, the anchor is always the first node in a subgraph, followed by its 1st-order neighbors, and then the 2nd-order neighbors, and so on; this is a natural partial ordering due to the "central status" of the anchor, under which we shall compare the point-clouds from different subgraphs. Under this partial order, we provide sufficient condition for the point clouds of two subgraphs to be different. We restrict ourselves to the case of equal-sized subgraphs, because two subgraphs with different numbers of nodes must have different point-clouds and density profiles.

Theorem 1: Consider unweighted, undirected subgraphs with no self-loops. Let $\mathbf{W}_1$ and $\mathbf{W}_2$ be the adjacency matrix of two equal-sized subgraphs, and w.l.o.g. assume the anchor node is the first node, and the random-walk scores starting from the anchor node are $\mathbf{p}_1$ and $\mathbf{p}_2$, respectively, for the two subgraphs. Let $\tilde{A}_1$ and $\tilde{A}_2$ be the transition matrix of the two subgraphs, and $\mathbf{B}_1^*$ and $\mathbf{B}_2^*$ be the adjugate matrix of $\mathbf{I} - c\tilde{A}_1$ and $\mathbf{I} - c\tilde{A}_2$, respectively. Then we have:

1) if $\frac{|\mathbf{B}_1^*[1,1]|}{\mathbf{B}_1^*[1,j]} \neq \frac{|\mathbf{B}_2^*[1,1]|}{\mathbf{B}_2^*[1,j]}$ for some $j \neq 1$, then $\mathbf{p}_1 \neq \mathbf{p}_2$;
2) if $\frac{|\mathbf{B}_1^*[1,1]|}{\mathbf{B}_1^*[1,j]} = \frac{|\mathbf{B}_2^*[1,1]|}{\mathbf{B}_2^*[1,j]}$ for all j, then $\mathbf{p}_1 = \mathbf{p}_2$;

here j is an integer in $[1, n]$.

Proof is in the Supplementary Materials. Theorem 1 shows that as long as the ratio between the first and the j-th entry in the first column of the adjugate matrix of $\mathbf{I} - c\tilde{A}_1^{-1}$ and $\mathbf{I} - c\tilde{A}_2^{-1}$ are different for at least one j ($2 \leq j \leq n$), then the point-clouds of the two enclosing subgraphs must be different. In order for the two point-clouds to be the same, the ratios have to be the same for all $2 \leq j \leq n$ across the two matrices. In fact, we speculate that as long as the two enclosing subgraphs are non-isomorphic, then their point-clouds will be different. We have empirically verified this conjecture on a large number of subgraph pairs. However, a strict proof involves the general problem of graph isomorphism, which can be quite challenging and is being pursued as a future research topic.

B. Smoothing Effect of the Kernel Density Estimator

In this subsection, we investigate how the bandwidth h in (4) influences the distance between two point-cloud density profiles, as measured by (6), with the potential to enhance the stability of the representation.

It's worth noting that the distance between two point-cloud distributions is influenced by various factors, including the locations of the points, the choice of Gaussian kernel centers, and the bandwidth. To specifically focus on the smoothing effect of the bandwidth so that explicit bounds can be derived, we confine our analysis to a simplified scenario. In this scenario, there is

only one non-overlapping pair of points across two point-clouds. Then the discrepancy between the two point-cloud distributions will be captured by the k Gaussian kernels located in the dipole plane, as quantified below.

Theorem 2: Let there be only one pair of non-overlapping points, $\mathbf{u}$ and $\mathbf{v}$, from two equal-sized point-clouds. The distance between $\mathbf{u}$ (or $\mathbf{v}$) and the k Gaussian centers $\boldsymbol{\mu}_j$'s are $\mathbf{d}_j^{\mathrm{u}} = \|\mathbf{u} - \boldsymbol{\mu}_j\|^2$ (or $\mathbf{d}_j^{\mathrm{v}} = \|\mathbf{v} - \boldsymbol{\mu}_j\|^2$) for $j = 1, 2, \ldots, k$. Let $\mathbf{d}_j^{\min} = \min(\mathbf{d}_j^u, \mathbf{d}_j^v)$, $\mathbf{d}_j^{\max} = \max(\mathbf{d}_j^u, \mathbf{d}_j^v)$, $\mathbf{d}_j^{\mathrm{uv}} = |\mathbf{d}_j^u - \mathbf{d}_j^v|$. The distance between the density profile $\mathbf{F}_1$ and $\mathbf{F}_2$ as defined in (6) from the two point-clouds is bounded by

$$\|\mathbf{F}_1 - \mathbf{F}_2\|^2 \geq \frac{1}{2\,h^2} \frac{\sum_j \exp\left(-\frac{\mathbf{d}_j^{\max}}{h^2}\right)}{\max\left(\sum_j \exp\left(-\frac{\mathbf{d}_j^u}{h^2}\right), \sum_j \exp\left(-\frac{\mathbf{d}_j^v}{h^2}\right)\right)},$$

$$\|\mathbf{F}_1 - \mathbf{F}_2\|^2 \leq \frac{1}{2\,h^2} \frac{\sum_j \exp\left(-\frac{\mathbf{d}_j^{\min}}{h^2}\right)}{\min\left(\sum_j \exp\left(-\frac{\mathbf{d}_j^v}{h^2}\right), \sum_j \exp\left(-\frac{\mathbf{d}_j^u}{h^2}\right)\right)}.$$

Proof is in the Supplementary Materials. Theorem 2 shows that the distance between two point-cloud densities, i.e., $\|\mathbf{F}_1 - \mathbf{F}_2\|$, is strongly modulated by the bandwidth h: a small h highlights their difference, while a large one trivializes it (due to the $\frac{1}{2\,h^2}$ term). An empirical verification can be found in the Supplementary Materials, in which we show that the curves of the derived bounds and the actual distance both drop with h for a data example satisfying the conditions in Theorem 2. In fact, even for more complex cases in which the two point-clouds have more than one pair of non-overlapping points, the difference between their density profile still decays with the bandwidth, as illustrated in the Supplementary Materials.

The relation between the distance $\|\mathbf{F}_1 - \mathbf{F}_2\|$ and the bandwidth is due to the smoothing effect of Gaussian kernels [47], since large bandwidth flattens distributions. More specifically, note that the j-th density feature $f(\boldsymbol{\mu}_j)$ is a statistical average on the "amount" of nodes in G_{xy} whose proximity to the two anchors is close to $\boldsymbol{\mu}_j = [\boldsymbol{\mu}_{j1}, \boldsymbol{\mu}_{j2}]$, and sum up to $|G_{xy}| = n_{xy}$. Therefore, if the links in a subgraph are perturbed (insertion or removal) but with nodes unchanged, the density $\mathbf{F}_{xy}$ is then a re-assignment of n_{xy} "density quota" among the k kernels, and will be relatively stable if smoothed by a Gaussian whose bandwidth exceeds the level of perturbation. This is desirable because real world graphs are always incomplete, so a reasonable amount of smoothing of the distances between different link-patterns can prevent the model from remembering (overfitting) the training graph too rigidly to generalize to unseen links. This is empirically verified in our ablation study reported in Section V-A3. The degree of smoothing can be determined adaptively through end-to-end optimization of the bandwidth parameter h, provided that a reasonable initial value is given.

V. Experimental Results

A. Link Prediction

We assess the performance of link-prediction algorithms based on three key criteria: (i) the Area Under the Curve (AUC) of link prediction, a widely used and highly interpretable metric in the field; (ii) the scalability of prediction algorithms; and (iii) the size, measured by the number of parameters, of the predictive models. Our evaluations encompass both artificial networks and real-world networks.

For artificial data, we generate networks of 1000 nodes using the Watts–Strogatz model and the Barabási–Albert (BA) model. The Watts–Strogatz model produces graphs with small-world properties [48]. It starts from a ring-shaped graph with $m = 1000$ nodes, where each node is connected with $k = 4$ neighbors on both sides. For every node, pick the $k = 4$ links connecting to its rightmost neighbors and rewire them (replace them with a random node) with probability p, while avoiding self-loops and duplication. We chose three sets of model parameters, namely $p = 0, 0.5, 0.8$, corresponding to regular ring, totally random network, and small-world network.

The Barabási–Albert (BA) model generates random scale-free networks with preferential attachment [49]. It begins with an initial network of m_0 nodes. New nodes are added one at a time. Each new node is connected to $m \leq m_0$ existing nodes with a probability p_i proportional to the number of links that the existing nodes already have, namely $p_i = k_i / \sum_j k_j$, where k_i is the degree of node i and the sum is made over all pre-existing nodes j. We chose $m = m_0 = 1, 3, 5$ as small integers so that the resultant degree distributions are scale free.

For real-world data, we have curated a collection of 13 benchmark networks widely employed in the link prediction community [15], [36], [38], [39]. This collection spans scientific, social, biological, and technological networks, showing diverse topological and statistical properties. The basic statistics for these networks are presented in Table III, and a brief introduction for each of the 13 networks is given below: Facebook [50]: social network from facebook with 4039 users and 88234 edges; USAir [51]: US air transportation network with 332 airports and direct flights; NetSci [52]: collaboration network among 1589 researchers from a variety of fields in network science; GRQ [53]: collaboration network from researchers of general relativity and quantum cosmology in arXiv from 1993 to 2003; Yeast [54]: a protein-protein interaction network in yeast with 2,375 nodes and 11,693 edges; Router [55]: a router-level Internet with 5,022 nodes and 6,258 edges; PPI [56]: protein-protein interaction in human tissues with 3890 nodes and 38292 edges; Power [57]: the power grid of the Western States of the U.S, with 4941 nodes (a generator, a transformator or a substation), and 6594 links (high-voltage power supply line); Citeseer [58]: citation network with 3312 scientific publications with 4732 links. Cora [58]: citation network with 2708 publications and 5429 links; Pubmed [58]: citation network with 19717 publications of diabete research with 44338 links; HPD [59]: protein-protein interaction about human protein in health and disease with 8756 nodes and 32331 edges; Email [6]: the email communication network at the University Rovira i Virgili in Tarragona in the south of Catalonia in Spain. Nodes are users and each edge represents that at least one email was sent.

Experimental results from nine competing methods are reported, including common neighbors [60], Jaccard [18], preferential attachment [61], Katz score [21], random-walk with restart [45], Adamic-Adar [19], and state-of-the-art GNNs like SEAL [36], Walk-Pooling [39], and Line-Graph [38].

TABLE I
AVERAGE LINK-PREDICTION AUC ON SYNTHETIC NETWORKS

Data	*WS-network*			*BA-network* ($m = m_0$)		
Alg\Para	$p = 0$	$p = 0.5$	$p = 0.8$	1	3	5
CN	99.51±0.01	61.46±0.56	51.16±0.58	49.54±0.01	53.96±0.46	57.17±0.63
Jaccard	99.58±0.01	61.58±0.57	51.17±0.58	49.02±0.27	53.67±0.42	55.90±0.55
PA	13.25±0.10	35.48±0.24	37.27±1.45	33.57±3.01	66.66±2.38	69.12±1.67
Katz	99.79±0.01	62.75±0.27	43.52±0.84	29.55±7.25	59.95±1.68	63.19±2.16
RWR	99.90±0.01	64.92±0.29	45.61±0.58	29.54±7.27	62.89±1.47	64.93±1.58
AA	99.51±0.01	61.54±0.55	51.15±0.56	49.54±0.01	54.04±0.46	57.17±0.71
SEAL	99.94±0.03	70.73±0.39	57.07±3.49	89.58±2.85	78.56±1.73	72.07±1.71
WP	99.83±0.09	74.87±0.50	59.81±1.51	95.59±0.52	81.25±0.56	72.56±1.72
LG	99.79±0.21	75.30±0.81	61.97±1.41	95.93±0.71	79.68±1.31	74.47±1.64
Ours	**99.99±0.01**	**75.58±0.19**	**62.18±1.55**	**96.30±1.08**	**85.13±0.85**	**76.21±1.93**

In WS-model, p the is rewiring probability, and the three values chosen are for regular, small-world, and more random networks, respectively; in BA-model (scale-free networks), m_0 and m are the initial # nodes and # attached nodes in each step.

TABLE II
AVERAGE LINK PREDICTION AUC FOR 10 LINK PREDICTION ALGORITHMS ON 13 POPULAR BENCHMARK NETWORKS; BEST AUCS ARE IN BOLD FONTS

Domain	*Social*		*Science Collaboration*		*Citation*		
Alg\Data	Facebook	Email	NetSci	GrQ	Citeseer	Pubmed	Cora
CN	99.27±0.03	84.97±0.72	93.12±0.91	91.96±0.25	67.21±0.46	65.42±0.26	73.64±0.69
Jaccard	99.08±0.02	84.57±0.71	90.38±1.12	92.25±0.35	63.75±0.36	65.24±0.28	73.10±0.33
PA	83.15±0.22	77.79±1.45	66.54±1.75	73.61±0.84	58.51±0.93	72.00±0.62	63.65±0.50
Katz	61.18±0.36	89.75±0.48	93.83±1.01	91.27±0.47	75.35±0.79	75.54±0.60	83.36±1.22
RWR	99.31±0.02	90.40±0.16	93.87±1.01	91.39±0.49	75.48±0.83	76.49±0.63	85.07±1.22
AA	99.38±0.02	85.00±0.72	85.25±0.99	91.99±0.25	65.18±0.57	64.87±0.22	75.51±0.80
SEAL	99.36±0.03	91.72±0.58	98.87±0.43	97.89±0.17	89.74±1.13	**97.15**±0.31	92.00±1.56
WP	99.65±0.02	93.01±0.12	98.49±0.70	98.46±0.01	89.48±0.02	**97.16**±0.01	92.62±0.01
LGLP	99.47±0.02	92.80±0.67	99.14±0.08	98.39±0.30	91.88±0.05	**97.15**±0.04	92.44±0.13
DSDN	**99.70 ±0.02**	**93.33±0.33**	**99.20±0.31**	**98.56±0.12**	**92.01±0.49**	**97.17±0.25**	**94.23±0.66**

Domain	*Biology*			*Infrastructure & Technology*		
Alg\Data	Yeast	PPI	HPD	USAir	Power	Router
CN	88.85±0.05	85.90±0.57	73.82±0.44	93.08±1.37	59.23±0.58	56.16±0.35
Jaccard	88.75±0.58	83.61±0.47	73.38±0.45	90.06±0.91	58.97±0.56	55.19±0.37
PA	82.31±0.87	88.47±0.35	82.19±0.40	88.67±1.57	44.50±0.73	48.01±1.17
Katz	91.98±0.77	89.75±0.40	86.36±0.35	92.39±1.61	64.77±1.48	38.69±0.97
RWR	92.61±0.75	91.14±0.36	87.64±0.38	94.44±1.55	65.17±1.53	38.75±0.96
AA	88.21±0.59	85.47±0.59	73.86±0.44	92.09±1.20	57.69±0.46	56.12±0.34
SEAL	97.53±0.18	**92.70±0.20**	93.59±0.27	96.08±0.58	87.77±0.87	96.37±0.74
WP	97.87±0.44	92.43±0.49	93.76±0.08	97.15±1.09	90.53±0.89	97.02±0.48
LGLP	97.62±0.03	92.40±0.08	93.16±0.30	97.09±0.14	85.40±0.22	95.66±0.09
DSDN	**98.06±0.27**	92.18±0.46	**93.87±0.09**	**97.65±0.75**	**92.52±0.47**	**97.19±0.52**

TABLE III
SUMMARY OF REAL-WORLD NETWORKS AND THEIR STATISTICS

Network	#nodes	#links	link ratio	avg degree	clus. coeff.
Facebook	4039	88234	0.0108	43.69	0.6055
USAir	332	2126	0.0386	12.8	0.6252
Cora	2708	5278	0.00144	3.90	0.2406
NetSci	1589	2742	0.00217	3.45	0.6377
Citeseer	3327	4552	0.00083	2.75	0.1414
PB	1222	16714	0.0224	27.4	0.3202
Pubmed	19717	44324	0.0023	4.50	0.0601
Yeast	2375	11693	0.00415	9.85	0.3057
PPI	3890	37845	0.005	19.5	0.1464
Router	5022	6258	0.0049	2.49	0.0115
Power	4941	6594	0.0054	2.70	0.0801
C.ele	297	2148	0.0487	14.5	0.2923
GRQ	5241	14484	0.00105	5.53	0.5297
BUP	105	441	0.080	8.40	0.4875
HPD	8756	32331	0.0084	7.38	0.1130
Email	1133	5451	0.0085	9.62	0.2203

1) Experimental Setting: The training and testing data are generated with the protocol widely used in link-prediction community [15], [36], [38], [39]. For an input network, we first randomly remove 10% observed links as the positive testing data, and randomly sample the same number of unconnected node pairs as negative testing data. The remaining 90% observed links and the same number of additionally sampled negative links are used as the positive and negative training data. We randomly generate 10 such splits for each network. All competing methods are evaluated on the same 10 data splits for each network, and the mean and the standard deviation of the AUCs are reported.

For our approach, we use 2-hop enclosing subgraphs, and the Gaussian centers $\{\boldsymbol{\mu}'_j s\}$ are initialized as follows: we first collect the enclosing subgraphs from the training links and project them as point clouds onto the 2-D dipole plane $[0, 1] \times [0, 1]$ with 0.05×0.05 grids, and then initialize $\boldsymbol{\mu}_j$'s as the centers of those grids with high point counts. One can also use the k-means clustering centers of the point-clouds to initialize $\boldsymbol{\mu}_j$'s with a pre-defined k (around 100 to 200). The bandwidth h is initialized at $h = 0.025$. The re-start probability c is chosen from $\{0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8\}$ using 5% of the training data as validation.[3] In other words, we encourage the

[3]Note that the validation set can also be used to choose other hyper-parameters, such as the number of hops, the initial bandwidth, and uniform-sampling interval used to initialize the Gaussian centers.

TABLE IV
AVERAGE MODEL SIZE OF OUR METHOD AND GNN-BASED ONES

Methods	Ours	SEAL [38]	WP [41]	LGLP [40]
Number of parameters	11k	56k	3,278k	21k
feature generation	0.3k	11k	12k	8k
fully-connected layers	10.7k	45k	3,266k	13k

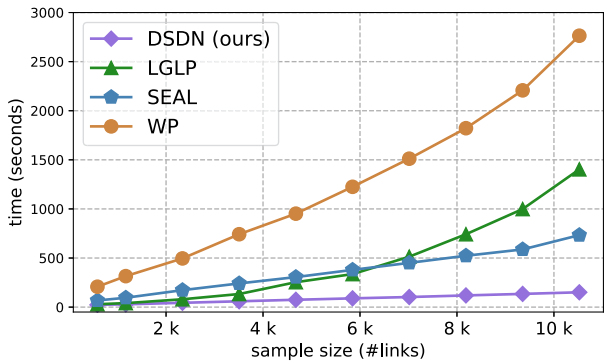


Fig. 3. Time consumption of GNN based link-prediction algorithms and our method versus the number of training links. Our approach scales linearly with sample size and is computationally more efficient.

random-walk to explore larger neighbors around the two anchor nodes of the target link. Three hidden FC-layers are used with dimension $(k, 64)$, (64,16), and (16,2). The model is optimized with Adam using an initial learning rate of 0.01 and a decay factor of 0.001.

For all the methods under evaluation, the focal link is removed from the enclosing subgraph of those positive link instances in the training set. The averaged model size for different models are listed in Table IV. The detailed calculations can be found in the Supplementary Materials.

2) Comparative Results: Tables I and II present the average Area Under the Curve (AUC) values with standard deviations over 10 random training/test splits for synthetic and real networks, respectively. On synthetic networks, our approach consistently achieves the highest AUC across various settings of model parameters, including regular, small-world, random, and scale-free networks. For real-world networks, our approach attains the highest AUC on 12 out of 13 networks, with improvements over GNN-based models ranging from 2% to 5% for challenging networks (e.g., Cora and Power). Notably, our model is approximately 300 times smaller than Walk-Pooling [39], the best GNN-based algorithm compared in this study (on average, as per Table IV). So the performance gains of our approach are considered encouraging, and clearly verify the effectiveness of the proposed link representation.

Along with the compact model size comes with superior computational efficiency: as shown in Fig. 3, our method is 5–20 times faster than GNNs. Experiments were also conducted on networks with node attributes, and the results can be found in the Supplementary Materials.

3) Impact of the Hyper-Parameters: In this subsection, we examine the impact of the hyper-parameters. Fig. 4(a) shows the impact of the jump-back probability $(1 - c)$ in the random-walk. As can be seen, when c grows larger and larger from 0.5 the performance of link prediction steadily improves and reaches a plateau. In fact, the optimal choice is around $c = 0.7$ for the networks investigated here, corresponding to a jump-back probability $1 - c = 0.3$. This means that on the one hand, the random-walk somehow remembers where it gets started (i.e., the anchor node x or y); on the other hand, it has the freedom to explore the whole subgraph, i.e., how the neighboring nodes of the link gather around the two anchors.

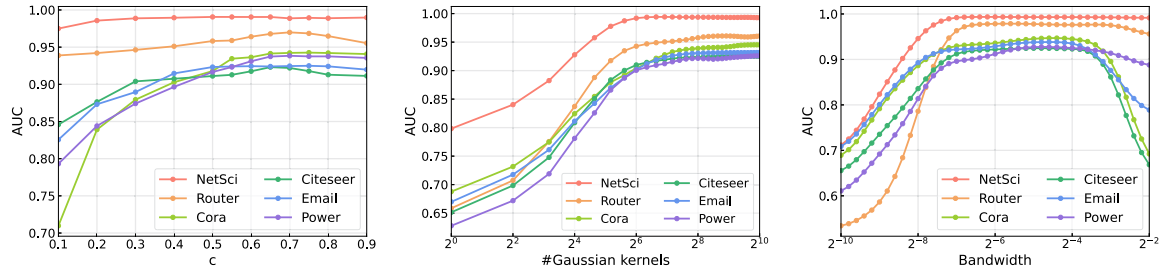
In Fig. 4(b), we present the AUC versus the number of Gaussian kernels k. It is evident that the AUC is relatively insensitive to the choice of k when it is sufficiently large. In our experiments, we find that the performance appears satisfactory when k is in the range from 100 to 200.

Fig. 4(c) shows the AUC of link prediction over the bandwidth h in (4). The bandwidth controls the degree of smoothing, which is beneficial in avoiding overfitting. Here the AUC is averaged over 5 random initialization of parameters. As can be seen, when the bandwidth increases from a small value to 0.05, predictive performance consistently improves, affirming the effectiveness of kernel smoothing in the proposed link representation scheme. However, when the bandwidth becomes too large, distinct point-cloud distributions may no longer be effectively discriminated, leading to expected drop in performance. In practice, the bandwidth parameter is optimized end-to-end given a reasonable initial value.

B. Cross-Domain Network Analysis

Cross-domain studies can be valuable in uncovering diversified link formation patterns from miscellaneous real-world networks and finding the underlying commonalities. However, establishing a universal representation for links across different networks remains challenging. On the one hand, GNN-based approaches such as SEAL [36] uses node2vec to extract topological features for each node, which results in network-specific link representations that cannot be directly compared across multiple networks. On the other hand, learning-based approaches are better suited for distinguishing between positive and NULL links, while our primary focus is on profiling the distribution of positive links across different networks.

The proposed link representation provides a universal framework for studying and comparing links across diverse fields simultaneously. First, the density profile (6) serves as a fixed-dimensional, well-aligned representation with consistent physical meaning even across networks (Section III-B). This is achieved by selecting the same set of Gaussian kernel centers $\boldsymbol{\mu}_j's$ in the dipole plane across different networks. Second, the link representation can be obtained in an unsupervised manner due to the strong inductive-bias introduced in our representation. Specifically, the kernel centers $\boldsymbol{\mu}_j$'s in the dipole plane can be chosen as the clustering centers of the aggregated point-clouds without using class labels. By doing this, the resulting link representation faithfully preserves the distribution of diverse patterns of positive links. This facilitates the construction of a global link-pattern atlas, through which we have uncovered



(a) The AUC of link prediction versus the choice of c that controls the re-start probability $(1-c)$. (b) The AUC of link prediction versus the choice of the number of Gaussian kernels k. (c) The AUC of link prediction versus the choice of the bandwidth parameter h.

Fig. 4. Impact of the jump-back probability $(1-c)$, the number of kernels k and the bandwidth parameter h on the performance of link prediction.

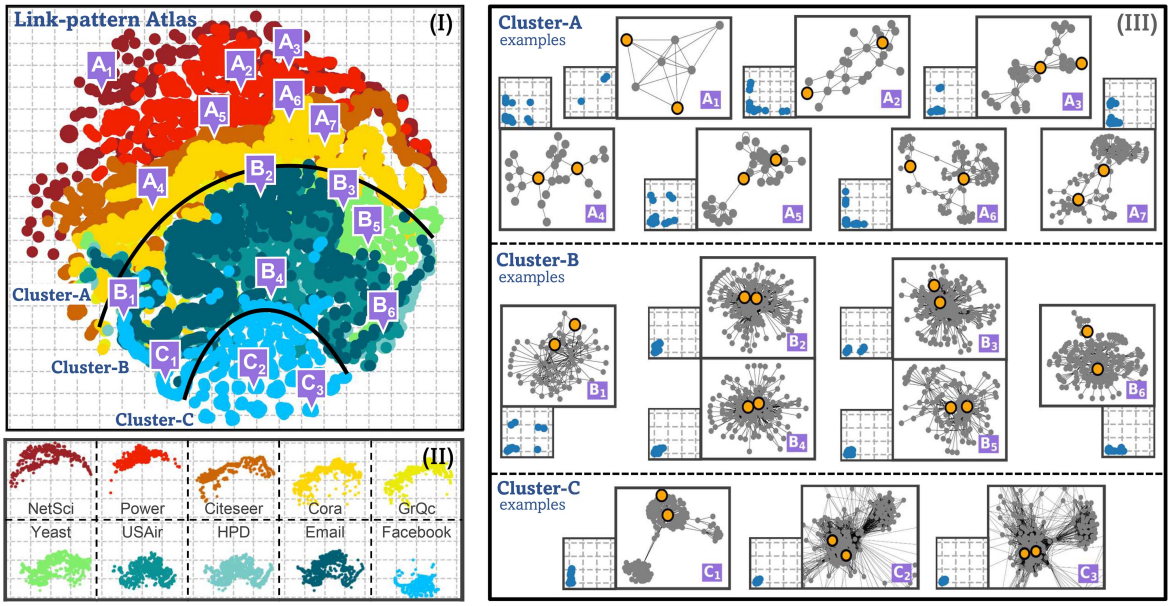


Fig. 5. Cross-domain study of network links collected from different science and engineering fields. Panel-I: link-pattern atlas including 30,000 links drawn randomly from 10 networks, embedded as 30,000 points by tSNE with one color for each network. Three main clusters are identified: **cluster-A: bridge style**, **cluster-B: radiation style**, and **cluster-C: community style**, each with a few representative link examples that are marked as A_1-A_7, B_1-B_6, C_1-C_3 for case study. Panel-II: the tSNE embedding in Panel-I is plotted separately for each network to better visualize the distribution of the link patterns in each network, allowing for a quick examination of the similarity between the networks. Panel-III: enclosing-subgraphs and point-clouds for 16 representative link examples; each subgraph is with gray nodes and two orange anchors; each point-cloud is with blue points and plotted in $[0, 0.3] \times [0, 0.3]$ of the dipole plane to better visualize details. Note that we have used 2-hop neighbors from each anchor node, so the resultant enclosing subgraph would allow a maximum pairwise distance up to 5-hops between the nodes. Section V provides detailed discussions.

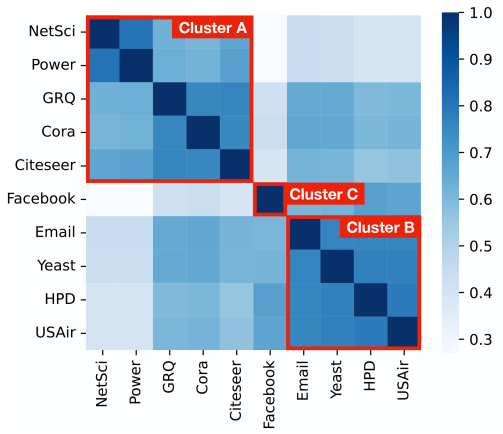


Fig. 6. Pairwise similarities of the 10 networks selected from Table I across 5 domains. The similarities are based on the distribution of the link-patterns from each network as measured through the proposed link representation.

intriguing common patterns of link formation and identified hidden network similarities extending beyond their original domains.

Here we report a preliminary study. We select 10 networks covering 5 different domains from Table II, and randomly sample 3000 true links from each network. By using 2-hop enclosing subgraphs, each link is represented as k-dimensional density profile as in (6) with $k = 64$, and further normalized by the average subgraph size from each network. Based on this representation, we then visualize the 30,000 links through their tSNE embedding, thus presenting a global link-pattern landscape in Fig. 5 (panel-I). Based on this embedding, we can see that the link patterns from the 10 networks form three visual clusters (though with overlaps), discussed below.

Cluster-A (top): NetSci (co-author, dark red), Power (red), Citeseer (citation, brown), Cora (citation, orange) and GrQ

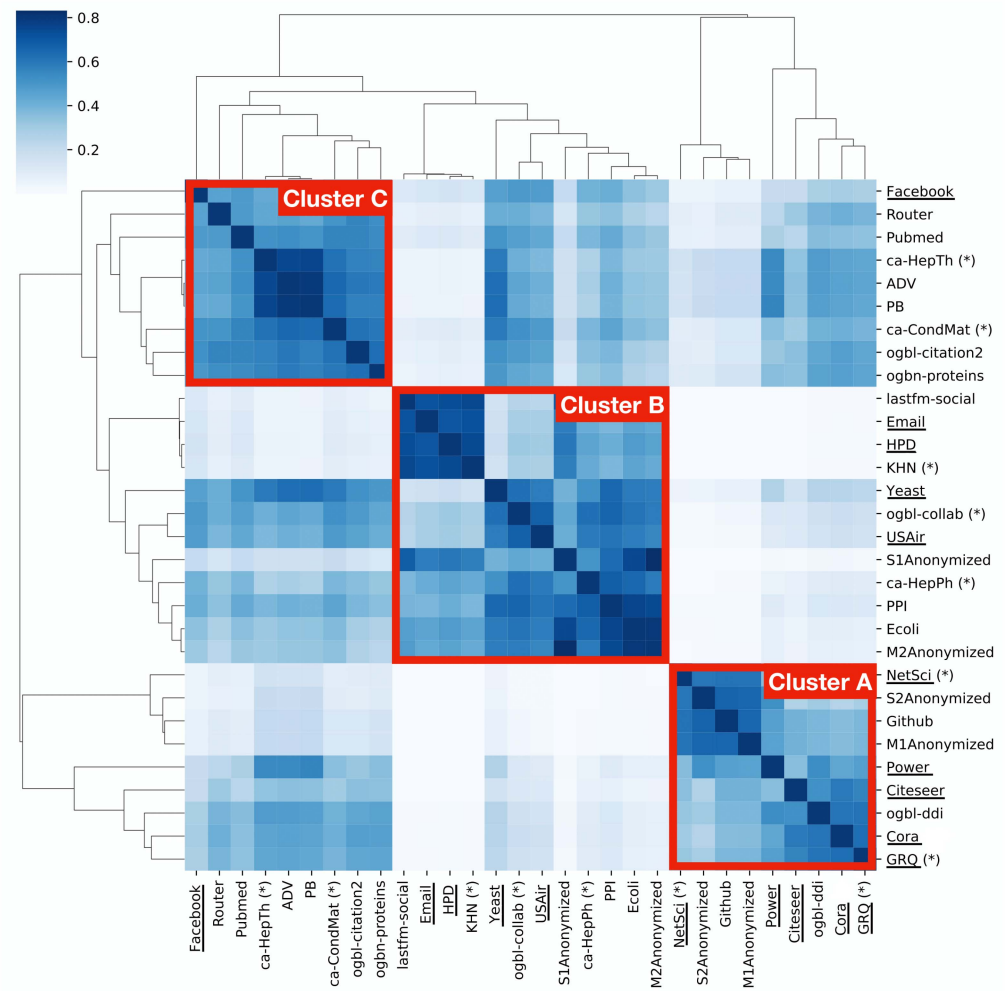


Fig. 7. Hierarchical clustering of 30 networks collected from a wide spectrum of disciplines. Three big clusters are identified and correspond to the three clusters A, B and C, as obtained in the Fig. 4 of the main text. The 10 networks with underlines are those used in the comparative studies in Section V-B. Networks marked with (*) are scientific networks that encode collaborative relations among the researchers.

(co-author, orange); the link distributions form a few zones (with overlaps) that are stacked layerwise in this cluster.

Cluster-B (middle): Email (dark green), protein networks Yeast (green) and HPD (green), and USAir (light green); the link distributions form curved manifold in this cluster.

Cluster-C (bottom): Facebook (blue); link distribution has many micro-clusters around the bottom of the landscape.

For visual clarity, links in each network are also plotted separately in Fig. 5 (panel-II). An intriguing observation is that networks from very different domains could exhibit strong similarity in the formation and characteristics of their link patterns, and group closely. Next, we elaborate on these clusters and look into some representative examples.

Bridge-style (Cluster-A): Many links in this cluster serve as a bridge to connect two nodes that are otherwise many hops away, indicating decentralized connections across distant subnetworks. For example, in Power network, typically a limited number of transmission lines are built to cover as many stations as needed under efficiency and cost considerations. Citation networks like Cora and Citeseer include many different research areas and so the links also spread out. Similarly, links in co-author networks like NetSci and GrQ are distributed, too. Intriguingly, we find that not all coauthor networks are necessarily dominated by decentralized link connections; instead, some coauthor networks may have more centralized collaboration similar to link patterns in Cluster-B, see the larger-scale case studies reported at the end of this section. We speculate the latter is due to the existence of a fraction of highly influential researchers in the area. The point-clouds in cluster A may either look L-shaped or break into a few small groups, see link examples A_1-A_6 in Fig. 5 (panel-III).

Radiation-Style (Cluster-B): Links in this cluster primarily connect a hub and a non-hub node or the central position of a star-shaped subnetwork to its periphery, exemplified by links B_2, B_3, B_5, and B_6. This pattern aligns with the organizational principles of networks falling within this cluster. For instance, in protein networks (Yeast and HPD), it is widely acknowledged that a small number of hub nodes play a central role in global network organization. In the USAir network, air flights are predominantly between one of the transfer stations and a terminal station. In the Email network (representing email communications from the users of Univ. Rovira i Virgili, Spain), many messages

are exchanged between a central unit (office, department head, group lead) and individuals (faculty, staff, student), following a hierarchical, self-similar pattern [6].

Community-Style (Cluster-C): Notably, links in this cluster are found inside communities, like C_1, C_2 and C_3. For link C_2 and C_3, the two anchors are structurally more symmetric inside the enclosing subgraph, and so the point-cloud spreads along $y = x$ in the dipole plane. In comparison, the two anchors of the links in cluster-B are structurally asymmetric. Another observation is that the enclosing subgraphs in Cluster C typically involve multiple communities for one target link, indicating a large number of communities extensively inter-connected with each other within a small number of hops in big social networks. In comparison, cluster B typically presents a single community in each enclosing subgraph.

Besides these representative cases, some transitional link patterns around cluster boundaries can also be found. For example, link B_1 is between cluster-B and cluster-C. Here, the two anchors of B_1 are structurally symmetric like cluster C, while the local enclosing subgraph only involves one community like cluster-B. For Link C_1, its two anchors are found inside a community where one is the bub node and the other is the boundary node, which is the key characteristic of cluster-B. On the other hand, the enclosing subgraph involves two communities, which is like cluster-C.

In Fig. 6 we report the network-level spectral clustering based on the link patterns from each network. Detailed procedures can be found in the Supplementary Materials. As can be seen, the similarity matrix clearly reveals three diagonal blocks, corresponding to the three clusters discussed above.

The cross-domain analysis based on the proposed link representation fosters a global understanding of how the link patterns from diversified application fields are distributed. It reveals intriguing network similarities that may otherwise be hidden. This universal platform could help promote new findings in network knowledge discovery, especially from the perspective of link-pattern characteristics or distributions. In Fig. 7 we use 30 networks from a wider variety of areas and examine their network-level grouping based on link-pattern distributions. The details of these networks can be found in the Supplementary Materials. A hierarchical clustering based on pairwise network similarities shows three dominant clusters, which are in accordance with the three clusters from the smaller-scale studies in Fig. 6.

Here we take a closer look at the 6 scientific collaboration networks: GRQ, NetSci, KHN, ogbl-collab, ca-ConcMat, and ca-HepTh. An interesting observation is that these 6 collaboration networks are not in the same cluster. For example, GRQ and NetSci are in Cluster A (bridge style), KHN and ogbl-collab are in cluster B (radiation-style), and ca-HepTh and ca-CondMat are in cluster C (community-style). This reflects the diversity of the link-pattern distributions of networks even for those from the same domain, due to the difference in the organization principles. We speculate that the diversity of the collaboration networks originates from the different ways of collaborations in different scientific communities.

VI. Conclusion

We have introduced a novel topological representation for link prediction. It transforms the enclosing subgraph of a target link into a fixed dimensional density distribution, which possesses desired expressiveness, interpretability, and cross-domain modelling power. Our experimental findings demonstrate that our method achieves a performance comparable or superior to that of GNN-based models, while simultaneously offering accelerated training speed and reduced model size. Moreover, our representation method establishes a versatile research platform for investigating cross-network topological information, which unravels interesting commonalities of link formation patterns across networks of different nature.

A number of interesting directions will be pursued in our future studies. For example, we will extend the proposed link representation to more general scenarios like (sub)graph representation learning. We will also apply it to large-scale case studies to facilitate domain-specific knowledge discovery like brain functional networks and protein interaction networks. Finally, we plan to study how the proposed link representation is related to network control, especially in building accurate predictive models that can map the distribution of the link patterns of a network to its global behaviour or functionality.

References

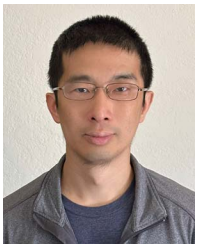
[1] L. A. N. Amaral, A. Scala, M. Barthelemy, and H. E. Stanley, "Classes of small-world networks," *Proc. Nat. Acad. Sci. USA*, vol. 97, no. 21, pp. 11 149–11 152, 2000.
[2] M. Newman, "The structure of scientific collaboration networks," *Proc. Nat. Acedemy Sci. USA*, vol. 98, no. 2, pp. 404–409, 2001.
[3] A.-L. Barabási, *Network Science*. Cambridge, U.K.: Cambridge Univ. Press, 2016.
[4] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. Hwang, "Complex networks: Structure and dynamics," *Phys. Rep.*, vol. 424, pp. 175–308, 2006.
[5] P. Holme and J. Saramäki, "Temporal networks," *Phys. Rep.*, vol. 519, no. 3, pp. 97–125, 2012.
[6] R. Guimerá, L. Danon, A. Díaz-Guilera, and F. A. A. Giralt, "Self-similar community structure in a network of human interactions," *Phys. Rev. E*, vol. 68, no. 3, 2006, Art. no. 065103.
[7] S. Laughlin and T. Sejnowski, "Communication in neuronal networks," *Science*, vol. 301, no. 5641, pp. 1870–1874, 2003.
[8] E. Hossain, Z. Han, and H. Poor, *Smart Grid Communications and Networking*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
[9] D. Mishra, T. Bepler, B. Teague, B. Berger, J. Broach, and R. Weiss, "An engineered protein-phosphorylation toggle network with implications for endogenous network discovery," *Science*, vol. 373, pp. 623–630, 2021.
[10] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A, Stat. Mechanics Appl.*, vol. 390, no. 6, pp. 1150–1170, Mar. 2011.
[11] A. Clauset, C. Moore, and M. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, pp. 98–101, 2008.
[12] R. R. Guimerá and S.-P. Marta, "Missing and spurious interactions and the reconstruction of complex networks," *Proc. Nat. Acad. Sci. USA*, vol. 106, pp. 22073–22078, 2009.
[13] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proc. Int. Conf. Inf. Knowl. Manage.*, 2003, pp. 556–559.
[14] J. Gao, S. V. Buldyrev, H. E. Stanley, and S. Havlin, "Networks formed from interdependent networks," *Nat. Phys.*, vol. 8, pp. 40–48, 2012.
[15] L. Lü, L. Pan, T. Zhou, Y.-C. Zhang, and H. E. Stanleya, "Toward link predictability of complex networks," *Proc. Nat. Acad. Sci. USA*, vol. 112, no. 8, pp. 2325–2330, 2019.
[16] A. Ghasemian, H. Hosseinmardi, A. Galstyan, E. M. Airoldi, and A. Clauset, "Stacking models for nearly optimal link prediction in complex networks," *Proc. Nat. Acad. Sci. USA*, vol. 117, no. 38, pp. 23393–23400, 2020.

[17] T. Zhou, L. Lü, and Y. Zhang, "Predicting missing links via local information," *The Eur. Phys. J. B*, vol. 71, no. 4, pp. 623–630, 2009.
[18] P. Jaccard, "Étude comparative de la distribution florale dans une portion des alpes et des jura," *Bull. del la Société Vaudoise des Sci. Nat.*, vol. 37, no. 142, pp. 547–579, 1901.
[19] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Social Netw.*, vol. 25, no. 3, pp. 211–230, 2003.
[20] S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual web search engine," *Comput. Netw.*, vol. 56, no. 18, pp. 3825–3833, 2012.
[21] W. R. Arney, "A refined status index for sociometric data," *Sociol. Methods Res.*, vol. 1, no. 3, pp. 329–346, 1973.
[22] C. Wang, V. Satuluri, and S. Parthasarathy, "Local probabilistic models for link prediction," in *Proc. IEEE 7th Int. Conf. Data Mining*, 2007, pp. 322–331.
[23] J. Neville, "Statistical models and analysis techniques for learning in relational data," *Comput. Sci. Dept. Fac. Publication Ser.*, 2006.
[24] K. Yu, W. Chu, S. Yu, V. Tresp, and Z. Xu, "Stochastic relational models for discriminative link prediction," *Adv. Neural Inf. Process. Syst.*, vol. 19, pp. 1553–1560, 2006.
[25] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computers*, vol. 8, pp. 30–37, 2009.
[26] P. W. Holland, K. B. Laskey, and S. Leinhardt, "Stochastic blockmodels: First steps," *Social Netw.*, vol. 5, pp. 109–137, 1983.
[27] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2014, pp. 701–710.
[28] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proc. 24th Int. Conf. World Wide Web*, 2015, pp. 1067–1077.
[29] A. Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2016, pp. 855–864.
[30] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proc. Int. Conf. World Wide Web*, 2010, pp. 641–650.
[31] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[32] M. Bronstein, J. Bruna, Y. LeCun, and A. Szlam, "Geometric deep learning: Going beyond Euclidean data," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 18–42, Jul. 2017.
[33] W. W. L. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: Methods and applications," in *Proc. Neural Inf. Process. Syst.*, 2017, pp. 1024–1034.
[34] J. Qiu, Y. Dong, H. Ma, J. Li, K. Wang, and J. Tang, "Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec," in *Proc. ACM Int. Conf. Web Search Data Mining*, 2018, pp. 459–467.
[35] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. 5th Int. Conf. Learn. Representations*, 2017, pp. 1–12.
[36] M. Zhang and Y. Chen, "Link prediction based on graph neural networks," in *Proc. 32nd Conf. Adv. Neural Inf. Process. Syst.*, 2018, pp. 5171–5181.
[37] M. Zhang, Z. Cui, M. Neumann, and Y. Chen, "An end-to-end deep learning architecture for graph classification," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 4438–4445.
[38] L. Cai, J. Li, J. Wang, and S. Ji, "Line graph neural networks for link prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5103–5113, Sep. 2022.
[39] L. Pan, C. Shi, and I. Dokmanić, "Neural link prediction with walk pooling," in *Proc. 10th Int. Conf. Learn. Representations*, 2022, pp. 1–12.
[40] A. Kumar, S. S. Singh, K. Singh, and B. Biswas, "Link prediction techniques, applications, and performance: A survey," *Physica A, Stat. Mechanics Appl.*, vol. 553, 2020, Art. no. 124289.
[41] V. Martínez, F. Berzal, and J. Cubero, "A survey of link prediction in complex networks," *ACM Comput. Surv.*, vol. 49, no. 4, pp. 1–33, Dec. 2016.
[42] G. Jeh and J. Widom, "Simrank: A measure of structural-context similarity," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2002, pp. 538–543.
[43] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
[44] B. Li, Y. Xia, S. Xie, L. Wu, and T. Qin, "Distance-enhanced graph neural network for link prediction," in *Proc. 38th Int. Conf. Mach. Learn.*, 2021, pp. 1–5.
[45] H. Tong, C. Faloutsos, and J. Pan, "Fast random walk with restart and its applications," in *Proc. 6th Int. Conf. Data Mining*, 2006, pp. 613–622.
[46] E. Parzen, "On estimation of a probability density function and mode," *Appl. Math. Lett.*, vol. 33, pp. 1065–1076, 1962.
[47] M. Wand, *Kernel Smoothing*. London, U.K.: Chapman and Hall/CRC, 1995.
[48] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
[49] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Rev. Modern Phys.*, vol. 74, no. 7, pp. 47–97, 2002.
[50] J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 539–547.
[51] V. Batagelj and A. Mrvar, "Pajek datasets," 2006. [Online]. Available: http://vlado.fmf.uni-lj.si/pub/networks/data/
[52] M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E*, vol. 74, no. 3, 2006, Art. no. 036104.
[53] J. Leskovec, J. M. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM Trans. Knowl. Discov. Data*, vol. 1, 2006, Art. no. 2.
[54] C. Mering et al., "Comparative assessment of large-scale data sets of protein–protein interactions," *Nature*, vol. 417, no. 6887, pp. 399–403, 2002.
[55] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, "Measuring ISP topologies with rocketfuel," *IEEE/ACM Trans. Netw.*, vol. 12, no. 1, pp. 2–16, Feb. 2004.
[56] Y. Qi, Z. Bar-Joseph, and J. Klein-Seetharaman, "Evaluation of different biological data and computational classification methods for use in protein interaction prediction," *Proteins, Struct., Function, Bioinf.*, vol. 63, no. 3, pp. 490–500, 2006.
[57] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world'networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
[58] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, and T. Eliassi-Rad, "Collective classification in network data," *AI Mag.*, vol. 29, no. 3, 2008, Art. no. 93.
[59] S. Peri et al., "Development of human protein reference database as an initial platform for approaching systems biology in humans," *Genome Res.*, vol. 13, pp. 2363–2371, 2003.
[60] G. Kossinets, "Effects of missing data in social networks," *Soc. Netw.*, vol. 28, pp. 247–268, 2006.
[61] A. L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

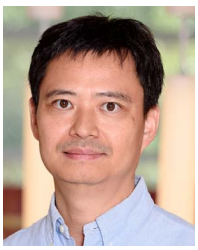
Kai Zhang received the PhD degree in computer science from the Hong Kong University of Science and Technology in 2008. He is currently a professor with the School of Computer Science and Technology, East China Normal University. His research focuses on neural networks, machine learning and their applications in bioinformatics, brain science, time series analysis, and complex networks. He is the recipient of NEC Labs America 2016 Business Contribution Award, and the KDD 2016 Best Paper Runner-up Award.

Junchen Shen received the bachelor's degree in applied physics from Nanjing University of Posts and Telecommunications in 2021. He is currently working toward the master's degree with the School of Computer Science and Technology, East China Normal University. His research interests include machine learning, graph neural networks, time series analysis, and recommendation systems.

Gaoqi He (Member, IEEE) received the PhD degree from the State Key Laboratory of CAD and CG, Zhejiang University, China, in 2007. He is currently a professor with the School of Computer Science and Technology, East China Normal University. His research interests include computer graphics, computer vision, and machine learning. His work aims to develop efficient and practical scene understanding and visual analysis algorithms with applications including video processing, crowd simulation, virtual reality, and augmented reality.



Yu Sun received the MS degree in electrical engineering from the University of Notre Dame in 2006 and the PhD degree in mechanical engineering from the University of Illinois, Urbana-Champaign in 2011. His research interests include machine learning, distributed optimization, game theory, reinforcement learning, and their applications.



Haibin Ling received the BS and MS degrees from Peking University in 1997 and 2000, respectively, and the PhD degree from the University of Maryland, College Park, in 2006. From 2000 to 2001, he was an assistant researcher with Microsoft Research Asia. From 2006 to 2007, he worked as a postdoctoral scientist with the University of California Los Angeles. In 2007, he joined Siemens Corporate Research as a research scientist; then, from 2008 to 2019, he worked as a faculty member with Temple University. In 2019, he joined Stony Brook University as a SUNY Empire Innovation professor with the Department of Computer Science. His research interests include computer vision, augmented reality, medical image analysis, machine learning, and human computer interaction. He received Best Student Paper Award at ACM UIST (2003), Best Journal Paper Award at IEEE VR (2021), NSF CAREER Award (2014), Yahoo Faculty Research and Engagement Award (2019), and Amazon Machine Learning Research Award (2019). He serves or served as associate editors for *IEEE Transactions on Pattern Analysis and Machine Intelligence* (PAMI), *IEEE Transactions on Visualization and Computer Graphics* (TVCG), *Computer Vision and Image Understanding* (CVIU), and *Pattern Recognition* (PR), and as area chairs for CVPR, ICCV, ECCV and WACV.

Hongyuan Zha received the BS degree in mathematics from Fudan University in 1984, and the PhD degree in scientific computing from Stanford University in 1993. He was a faculty member of College of Computing with Georgia Institute of Technology from 2006 to 2020, and the Department of Computer Science and Engineering, Pennsylvania State University from 1992 to 2006. His current research interest lies in machine learning and its applications. He has won many prominent academic awards including Leslie Fox Prize (second prize,1991) awarded by the Institute of Mathematics and Applications (IMA), the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2011) Best Student Paper Award (as advising Professor), the 26th NeurIPS Outstanding Paper Award (2013). He is a X.Q. Deng Presidential chair professor of The Chinese University of Hong Kong, Shenzhen and the executive dean of the School of Data Science.

Honglin Li received the bachelor's degree in fine chemistry, and the PhD degree in engineering mechanics from the Dalian University of Technology, in 2000 and 2005, respectively. As a post-doctor, he studied with the Shanghai Institute of Materia Medica from 2005 to June 2007. Since 2008, he worked with the School of Pharmacy in East China University of Science and Technology as a professor of Medicinal Chemistry and Computational Chemistry. Currently, he is the director of the Innovation Center for AI and Drug Discovery, East China Normal University. He has received a number of academic awards, including: the 13th Science and Technology Award for Chinese Youth (2013), National Youth Science and technology innovation leader (2015), Ten Thousand Talent Program Leading scientist (2017), National Science Fund for Distinguished Young Scholars (2018). His is interested in novel computational tools to rationalize and improve the efficiency, speed and cost-effectiveness of the drug discovery process. He has made some innovative achievements by combining dry/wet research approaches in drug design, drug interaction mechanism investigation and target identification.

Jie Zhang received the PhD from the Hong Kong Polytechnic University in 2008 and subsequently held a post-doctoral fellowship at the same institution from 2008 to 2010. Currently, a professor with the Institute of Brain-inspired Intelligence, Fudan University, his research focuses on complex and dynamic systems, computational neuroscience, and brain disorders. His work is dedicated to developing interdisciplinary computational approaches to shed light on the principles of brain information processing and functional network organizations underlying various psychiatric disorders. He was a finalist for the 2007 Young Scientist Award in the field of physical/mathematical sciences.