PHLP: Sole Persistent Homology for Link Prediction - Interpretable Feature Extraction

Junwon You, Eunwoo Heo, Jae-Hun Jung

Abstract— Link prediction (LP), inferring the connectivity between nodes, is a significant research area in graph data, where a link represents essential information on relationships between nodes. Although graph neural network (GNN)-based models have achieved high performance in LP, understanding why they perform well is challenging because most comprise complex neural networks. We employ persistent homology (PH), a topological data analysis method that helps analyze the topological information of graphs, to explain the reasons for the high performance. We propose a novel method that employs PH for LP (PHLP) focusing on how the presence or absence of target links influences the overall topology. The PHLP utilizes the *angle hop subgraph* and new node labeling called *degree double radius node labeling (Degree DRNL)*, distinguishing the information of graphs better than DRNL. Using only a classifier, PHLP performs similarly to state-of-the-art (SOTA) models on most benchmark datasets. Incorporating the outputs calculated using PHLP into the existing GNN-based SOTA models improves performance across all benchmark datasets. To the best of our knowledge, PHLP is the first method of applying PH to LP without GNNs. The proposed approach, employing PH while not relying on neural networks, enables the identification of crucial factors for improving performance.

Index Terms—Graph analysis, link prediction, persistent homology, topological data analysis.

I. INTRODUCTION

GRAPH data pervade numerous domains such as social networks, biological systems, recommendation engines, and e-commerce networks [1], [2]. The graph is well-suited for modeling complex real-world relationships.

Predicting missing or potential connections within a graph is essential for many applications, unlocking valuable insight and facilitating intelligent decision-making. The ability to predict future network interactions can be applied to diverse domains, including friend recommendations on social networks [3]–[5], knowledge graph completion [6], [7], identification of potential drug-protein interactions in bioinformatics [8], [9], prediction protein interactions [9]–[11], and optimization of supply chain logistics [12], [13].

The link prediction (LP) problem has been categorized into three major paradigms: heuristic methods, embedding methods, and graph neural network (GNN)-based methods, which are explored in detail in Section II. Recently, compared

Junwon You and Eunwoo Heo are co-first authors.

Junwon You, Eunwoo Heo, and J.-H. Jung are with the Department of Mathematics, Pohang University of Science and Technology (POSTECH), Pohang, South Korea. J.-H. Jung is also with the Graduate School of Artificial Intelligence, POSTECH, Pohang, South Korea.
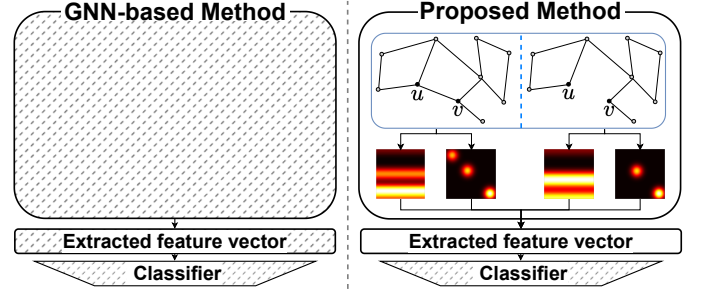
Fig. 1. Difference between the GNN-based and proposed methods. (Left) The GNN-based method extracts feature vectors through optimization (dashed area), making it difficult to interpret what these vectors represent. (Right) The proposed method extracts feature vectors through the designed analysis process, resulting in interpretable vectors.

to heuristic [3], [14]–[18] and embedding methods [19]–[22], GNN-based models have achieved significant score improvements in capturing intricate relationships within graphs [23]–[28].

However, GNN-based methods are comprised of neural networks, making it challenging to understand the reasons for their performance. To explore these reasons, we employ persistent homology (PH), a mathematical tool in topological data analysis (TDA) that enables the inference of topological information regarding the manifold approximating the data [29], [30] by quantifying the persistence of topological features across multiple scales. Various research has had successful outcomes in applying PH to graph classification and node classification tasks [31]–[40]. In contrast, relatively few studies have explored using PH for LP. The topological loop-counting (TLC) GNN [27] is a notable example that uses PH. The TLC-GNN injects topological information into a GNN, and experiments were conducted on benchmark data where node attributes are available.

In this context, as illustrated in Fig. 1, we present a novel approach to LP, called PHLP, which calculates the topological information of a graph. To use the topological information of subgraphs for LP, we measure how the topological information changes depending on the existence of the target link, as illustrated in Fig. 2. To extract topological information from various perspectives, we utilize *angle hop subgraphs* for each target node. Additionally, we propose new node labeling called *degree double radius node labeling (Degree DRNL)*, which incorporates degree information for each node, using DRNL [24].

The contributions are summarized as follows:
- We develop an explainable LP method, PHLP, that em-

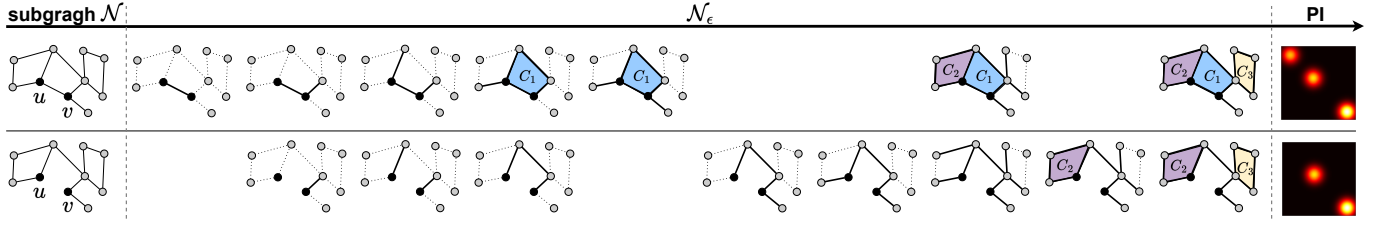**subgragh** $\mathcal{N}$ ⋮ $\mathcal{N}_\epsilon$ **PI**

Fig. 2. Topological features in subgraphs with and without a target link (u, v). The diagram illustrates the topological information extraction process for the subgraph $\mathcal{N}$, as described in Section III-D. The presence (top) or absence (bottom) of the target link changes the topological structure of the graph. Top row: When the target link is connected, three features (C_1, C_2, and C_3) are detected shown in the persistence image (PI) in the right column. The PI represents the topological features of the subgraph $\mathcal{N}$ (Section III-E). Bottom row: When the target link is absent, only two features (C_2 and C_3) are detected as depicted in the corresponding PI.

ploys the topological information for LP through PH without relying on neural networks, as illustrated in Fig. 1.

- We demonstrate that the proposed method, even with a simple classifier such as a multilayer perceptron (MLP), can achieve LP performance close to that of state-of-the-art (SOTA) models. This method surpassed the SOTA performance for the Power dataset.
- We reveal that merely incorporating vectors computed by PHLP into existing LP models, including SOTA models, can improve their performance.
- To the best of our knowledge, the proposed method using PH without a GNN is the first to achieve performance close to that of SOTA models.

II. RELATED WORK

A. Link Prediction

Heuristic Methods. Heuristic-based approaches to LP compute the predefined structural features within the observed nodes and edges of the graph. Classic methods, such as common neighbors [3], Adamic-Adar [3], Jaccard coefficient [14], and preferential attachment [15], rely on simple heuristics that capture certain aspects of node relationships. Zhou *et al.* [16] proposed a local random walk method, whereas Jeh and Widom [18] developed SimRank to quantify similarity based on the structural context. Although heuristic methods provide a preliminary understanding of LP, they are limited by their inability to capture complex relationships within graphs. Furthermore, heuristic methods are effective only when the defined heuristics align with the graph structure; therefore, applying heuristic methods across all graph datasets can be challenging.

Embedding Methods. Embedding methods map nodes from the graph into a low-dimensional vector space where geometric relationships mirror the graph structure. Koren *et al.* [19] demonstrated the power of matrix factorization for collaborative filtering. Perozzi *et al.* [20] introduced DeepWalk, using random walks to generate node sequences and employing the skip-gram model to produce embeddings. Tang *et al.* [22] developed large-scale information network embedding (LINE), which preserves local and global structures. Grover and Leskovec [21] further advanced this approach with Node2Vec (N2V), proposing a flexible notion of the neighborhood to capture diverse node relationships.

Embedding methods are advantageous due to their applicability regardless of the data characteristics using optimization. Node representations capture global properties and long-range effects through the learning process. However, these methods often require significantly large dimensions to express basic heuristics, resulting in lower performance than heuristic methods [41]. Moreover, in embedding methods, Ribeiro *et al.* [42] explained that two nodes with similar neighborhood structures may have vastly different embedded vectors, especially when they are far apart in the graph, leading to incorrect predictions.

GNN-Based Methods. The GNN has become a pivotal approach to LP due to its ability to grasp graph-structured data. By effectively incorporating local and global information through message passing and graph aggregation layers, GNNs enhance LP performance. The model by Zhang *et el.* [24] uses subgraphs as the primary structural units to learn and predict connections, resulting in significant improvement. This paradigm shift led to research focusing on refining and advancing subgraph methods in the context of GNNs [25], [26], [28]. Following this trend, Pan *et al.* [28] proposed WalkPool (WP), a new pooling mechanism that uses attention to jointly encode node representations and graph topology into learned topological features. However, despite their superior performance, GNN-based methods pose a challenge in comprehending the underlying mechanisms driving their predictions. Within this context, we develop the PHLP, based on PH, with performance comparable to GNN-based models.

B. Persistent Homology on Graph Data

In recent years, PH, a method of analyzing the topological features of data, has been widely used to analyze graph data. It has demonstrated its effectiveness in graph classification tasks by analyzing the topology of graphs [31]–[38] and has been applied to node classification tasks [31], [39], [40]. However, its suitability for LP tasks has been limited, and research on applying PH for LP has progressed slowly. Yan *et al.* [27] proposed an intriguing approach by integrating PH with GNNs. While their model demonstrates the potential of PH for capturing topological features of graph data, it relies on GNN structures. Additionally, the TLC-GNN requires further research on datasets without node attributes.

Although PH has demonstrated success in graph and node classification tasks, its filtration technique, tailored to analyzing the entire graph structure, might not be optimal for LP

(a) (k,l)-PHLP with Classifier
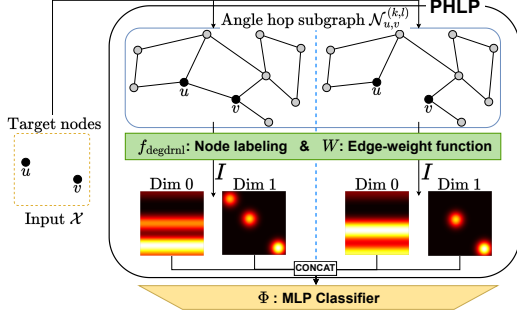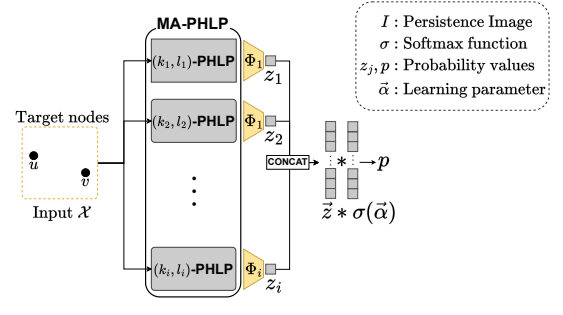

(b) MA - PHLP with Classifier

Fig. 3. Overall structure of persistent homology for link prediction (PHLP) and multiangle PHLP (MA-PHLP). (a) PHLP calculates the topological information based on the existence of target links in angle hop subgraphs for each target node. (b) With a classifier, MA-PHLP integrates topological information across various angles to perform LP.

as the role of each node in LP differs from that in graph or node classification tasks. To address this challenge and advance research in LP, we develop a filtration method tailored explicitly to LP tasks.

III. METHOD

A. Outline of the Proposed Methods

We propose (a) PHLP and (b) multiangle PHLP (MA-PHLP) as described in Fig. 3. The PHLP method analyzes the topological structure of the graph, focusing on target links. First, PHLP samples a (k,l)-angle hop subgraph for the given target nodes (Section III-B). Then, PHLP computes persistence images (PIs; Section III-E) for cases with and without the target link. To calculate PIs, we introduce the node labeling and define the edge-weight function (Section III-C). Through PHLP, each target node is transformed into a vector comprising PIs. In addition, LP is performed using the calculated vectors with a classifier (Section III-F). To reflect diverse topological information, we also propose MA-PHLP, which analyzes data from various angles (Section III-G).

B. Extracting Angle Hop Subgraph

Given a graph $G = (V, E)$ and two nodes $u, v \in V$, a k-hop enclosing subgraph for (u, v) is defined as $\mathcal{N}_{u,v}^k = (V', E')$ such that

$$V' = \{z \in V \mid d(u, z) \leq k \text{ or } d(z, v) \leq k\},$$
$$E' = \{(z, w) \in E \mid z \in V' \text{ and } w \in V'\},$$

where $d(z, w)$ is the minimum number of edges in any path from z to w in G. We define a (k, l)-angle hop enclosing subgraph, where the term "angle" signifies viewing the subgraph from multiple perspectives. The (k,l)-angle hop subgraph is a generalization of the k-hop subgraph. Given a graph $G = (V, E)$ and two nodes $u, v \in V$, a (k, l)-angle hop enclosing subgraph for (u, v) is defined as $\mathcal{N}_{u,v}^{(k,l)} = (V', E')$ such that

$$V' = \{z \in V \mid d(u, z) \leq k \text{ or } d(z, v) \leq l\},$$
$$E' = \{(z, w) \in E \mid z \in V' \text{ and } w \in V'\}.$$

Thus, the angle hop can generate subgraphs in various forms, providing flexibility to adapt to various graph characteristics.

C. Filtration of the Subgraph

For a given subgraph, the Rips filtration [43]–[45] is employed to calculate the topology using PH. To apply the Rips filtration, we define an edge-weight function using node labeling that reflects the topology of the given graph.
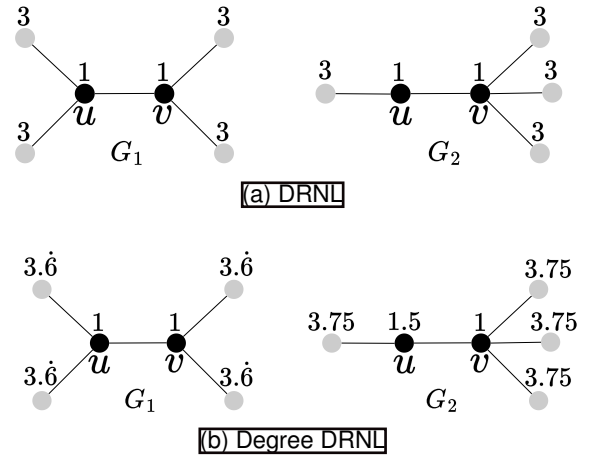
Fig. 4. Node labeling on graphs. (a) Node label values without considering the graph structure cannot distinguish between G_1 and G_2 using DRNL. (b) Applying Degree DRNL allows G_1 and G_2 to be distinguished solely by node label values.
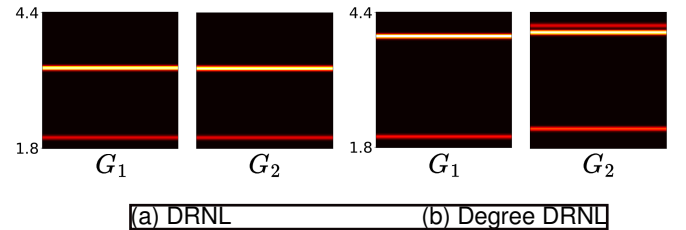
Fig. 5. Persistence images (PIs) for two node labeling methods for the graphs in Fig. 4. (a) DRNL exhibits identical zero-dimensional PIs for G_1 and G_2, (b) Degree DRNL produces distinct outcomes, effectively distinguishing between the two.

Degree DRNL. Zhang *et al.* [24] introduced DRNL, which computes the distance from any node to two fixed nodes. For any subgraph $\mathcal{N} = (V', E')$ of G and two nodes $a, b \in V'$,

the DRNL $f_{\text{drnl}}^{(a,b)} : V' \to \mathbb{N}$ based on (a, b) of G for any vertex w in V', is defined as

$$f_{\text{drnl}}^{(a,b)}(w) = 1 + \min(d(w,a), d(w,b)) + q_w(q_w + r_w - 1),$$

where $q_w \in \mathbb{Z}$ and $r_w \in \{0, 1\}$ are integers representing the quotient and remainder, respectively, such that $d(w, a) + d(w, b) = 2q_w + r_w$. We call these two nodes, a and b, *center nodes*. These center nodes do not need to be the target nodes used when extracting the subgraph.

However, DRNL encounters limitations when the graph is transformed into node-label information. As depicted in Fig. 4a, DRNL assigns the same node labels to different graphs, resulting in identical zero-dimensional PIs (Fig. 5a, Section III-E). To incorporate the local topology of each node with the effects of DRNL, we introduced *Degree DRNL*. For a given subgraph $\mathcal{N} = (V', E')$ of G and center nodes $a, b \in V'$, the Degree DRNL $f_{\text{degdrnl}}^{(a,b)} : V' \to \mathbb{R}$ based on (a, b), for all vertex w in V', is defined as

$$f_{\text{degdrnl}}^{(a,b)}(w) = f_{\text{drnl}}^{(a,b)}(w) + \frac{M - \deg(w)}{M},$$

where M denotes the maximum degree of nodes in $\mathcal{N}$. The $(M - \deg(w))/M$ term above assigns larger values for lower degrees of w. When $M = \deg(w)$, the value of Degree DRNL matches the original DRNL, ensuring that the edges connected to nodes with higher degrees are assigned smaller values, promoting their earlier emergence in the filtration. Fig. 4b demonstrates various node labels obtained using Degree DRNL, resulting in PIs that can be distinguished from each other (Fig. 5b).

Edge-weight function. For a given subgraph $\mathcal{N} = (V', E')$, $f : V' \to \mathbb{N}$ denotes any node labeling function. The edge-weight function $W : E' \to \mathbb{R}$, for any edge (w, z) in E', is defined as

$$W(w, z) = \max(f(w), f(z)) + \frac{\min(f(w), f(z))}{\max(f(w), f(z))}.$$

The min/max term in the definition of W refines values further, enhancing the discriminative power by reducing the occurrence of identical edge weights.

D. Persistent Homology

Given an edge-weighted subgraph $\mathcal{N} = (V', E', W)$, we construct a Rips filtration and compute its PH. First, we create a sequence of subgraphs $\{\mathcal{N}_\epsilon\}_{\epsilon \in \mathbb{R}}$, where each $\mathcal{N}_\epsilon = (V', E'_\epsilon)$ and $E'_\epsilon = \{e \in E \mid W(e) \le \epsilon\}$. Second, we convert each subgraph $\mathcal{N}_\epsilon$ into the Rips complex $K_\epsilon = \{\tau \in \mathbb{X} \mid (w, z) \in E'_\epsilon$ for any two vertices $w, z \in \tau\}$, where $\mathbb{X}$ is the power set of V'. In K_ϵ, a simplex τ is formed when the vertices in τ are pairwise connected by edges in $\mathcal{N}_\epsilon$. Then, the Rips filtration is obtained as $K_{\epsilon_1} \hookrightarrow K_{\epsilon_2} \hookrightarrow \cdots \hookrightarrow K_{\epsilon_m} = \mathbb{X}$ for $\epsilon_1 \le \epsilon_2 \le \cdots \le \epsilon_m$. Third, we compute the p-dimensional homology group $H_p(K_\epsilon)$ for each complex K_ϵ and track how these groups change as ϵ increases. The persistence diagram D [45] comprises persistence pairs (b, d) representing the ϵ values at which a homological feature appears b and disappears d, respectively, in the filtration.

E. Persistence Image

We convert the persistence diagram into a PI [46]. For a given persistence diagram D, consider a linear transform $L : \mathbb{R}^2 \to \mathbb{R}^2$ defined by $L(x, y) = (x, y - x)$. The image set of D under this transformation is denoted as $L(D)$. For each point (b, d') in $L(D)$, a weight function $\phi_{(b,d')} : \mathbb{R}^2 \to \mathbb{R}$ is defined that assigns a weight to each point in the persistence diagram. A common choice for $\phi_{(b,d')}$ is the Gaussian function centered at (b, d'). The nonnegative function is defined as $h : \mathbb{R}^2 \to \mathbb{R}$, as $h(x, y) = 1/\log(1 + |y|)$. The function h is zero along the horizontal x-axis, and is continuous and piecewise differentiable, satisfying the conditions presented in [46]. The persistence surface $\rho_D : \mathbb{R}^2 \to \mathbb{R}$ is defined as

$$\rho_D(z) = \sum_{(b,d') \in L(D)} h(b, d')\phi_{(b,d')}(z).$$

The continuous surface ρ_D is discretized into a finite-dimensional representation over a predefined grid. This grid consists of n cells, each corresponding to a specific region in the plane. The PI is defined as an array of values $I(\rho_D)_p$ for each cell p. Each $I(\rho_D)_p$ in this array is computed by integrating the persistence surface ρ_D over the area of cell p:

$$I(\rho_D)_p = \iint_p \rho_D \, dy \, dx.$$

F. Predicting the Existence of the Target Link

For the given target nodes (u, v), we sample the (k, l)-angle hop subgraph $\mathcal{N}_{u,v}^{(k,l)}$, denoted as $\mathcal{N}^-$ (Section III-B), assuming that the target link does not exist during this process. On this subgraph, we extract topological features by calculating PH and its vectorization (i.e., the PI, as described in Sections III-D and III-E). The vectorization is calculated for each dimension and concatenated. If $k \ne l$, for symmetry, we repeat the same process with the (l, k)-angle hop subgraph once and consider the average of the two vectors, denoting this vector as x^-. To observe the difference in topological features, we consider a subgraph $\mathcal{N}+$ obtained by connecting the target link to $\mathcal{N}^-$. For this graph, x^+ denotes the vector obtained using this method.

To predict the existence of the target link with the vectors x^- and x^+, we employ an MLP classifier $\Phi : \mathbb{R}^{2(d+1)n^2} \to \mathbb{R}$ where n represents the resolution of the PI, and d denotes the maximal dimension of PH. The model predicts the existence of a link between two target nodes with the following probability:

$$z_{uv} = \sigma(\Phi(x)),$$

where x is the concatenation of x^- and x^+, and σ is the activation function. For the training dataset $\mathcal{X} \subseteq V \times V$, comprising positive and negative links corresponding to the elements of E and $(V \times V) \setminus E$, respectively, we define the loss function as follows:

$$\sum_{(u,v) \in \mathcal{X}} BCE(z_{uv}, y_{uv}),$$

where $BCE(\cdot, \cdot)$ represents the binary cross-entropy loss and y_{uv} denotes the label of the target link (u, v), which is 0 for negative links or 1 for positive links.

G. Multiangle PHLP

The MA-PHLP maximizes the advantages of PHLP by examining data from various angles through the extraction of subgraphs based on a hyperparameter, the maximum hop (max hop, denoted as H). The types of angles are elements of all combinations of k and l within the set $\{(k,l) \in \mathbb{Z}^2 | 0 \leq l \leq k \leq H, k > 0\}$. If we define the prediction probability of a PHLP for each type of angle hop as z_i for $i = 1, 2, ..., N$, then MA-PHLP predicts the likelihood of the link existence with the following probability:

$$p = \sum_{i=1}^{N} \alpha_i z_i,$$

where $\alpha = (\alpha_1, ..., \alpha_N) \in \mathbb{R}^N$ is a trainable parameter. We apply the softmax function to the parameter α to ensure that the sum of all elements equals 1. Moreover, MA-PHLP is trained using the binary cross-entropy loss.

H. Hybrid Method

The proposed approach easily integrates with existing subgraph methods. Subgraph methods treat the LP task as a binary classification problem comprising two components: a feature extractor F and classifier P. Vectors with PH information calculated using the proposed methods are incorporated through concatenation before the classifier. The detailed process of the hybrid method is outlined as follows:

1) **Subgraph Extraction:** For the given graph G and target nodes (u, v), k-hop subgraph $\mathcal{N}_{u,v}^k$ is extracted.
2) **Feature Extraction:** Existing methods extract features $Z = F(\mathcal{N}_{u,v}^k)$ from the subgraph.
3) **Persistent Image Calculation:** The methods described in Sections III-C, III-D, and III-E are applied to $\mathcal{N}_{u,v}^k$, where I denotes the PI vector. An MLP $\Phi : \mathbb{R}^m \to \mathbb{R}^n$ transforms the PI into a format similar to Z. For the hybrid method of MA-PHLP, $\mathcal{N}_{u,v}^k$ is replaced with multiangle subgraphs, concatenating their PI vectors.
4) **Classification:** Next, $\alpha_1 Z$ and $\alpha_2 \Phi(I)$ are concatenated, where α_1 and α_2 are trainable parameters. The softmax function is applied to the parameter $\alpha = (\alpha_1, \alpha_2)$, ensuring that the sum of elements equals 1, denoted by J. This concatenated vector is classified using the existing method's classifier, $P(J)$.

IV. EXPERIMENTS

This section evaluates the performance of MA-PHLP. The experiments were also conducted using only zero-dimensional homology (MA-PHLP (dim0)). We used the area under the curve (AUC) [47] as an evaluation metric. We repeated all experiments 10 times and reported the mean and standard deviation of the AUC values.

A. Experimental Settings

Baselines. To evaluate the effectiveness of PHLP, we compared the proposed model with five heuristic methods, four embedding-based methods, and two GNN-based models. The heuristic methods include the Adamic-Adar (AA) [3], Katz index (Katz) [48], PageRank (PR) [49], Weisfeiler-Lehman graph kernel (WLK) [50], and Weisfeiler-Lehman neural machine (WLNM) [51]. For the embedding-based methods, we applied N2V [21], spectral clustering (SPC) [52], matrix factorization (MF) [19], and LINE [22]. Moreover, SEAL [24] and WP [28] represent the GNN-based methods.

Datasets. In line with previous studies [24] and [28], we eval-

TABLE I
STATISTICS OF THE DATASETS

Dataset	#Nodes	#Edges	Avg. node deg.	Density
USAir	332	2126	12.81	3.86e-2
NS	1589	2742	3.45	2.17e-3
PB	1222	16714	27.36	2.24e-2
Yeast	2375	11693	9.85	4.15e-3
C.ele	297	2148	14.46	4.87e-2
Power	4941	6594	2.67	5.40e-4
Router	5022	6258	2.49	4.96e-4
E.coli	1805	15660	16.24	9.61e-3

uate the performance of our MA-PHLP on the eight datasets in Table I without node attributes: USAir [53], NS [54], PB [55], Yeast [56], C. elegans (C. ele) [57], Power [57], Router [58], and E. coli [59]. The detailed statistics for each dataset are summarized in Table I.

Implementation Details. All edges in the datasets were split into training, validation, and testing datasets with proportions of 0.85, 0.05, and 0.1, respectively, ensuring a fair comparison with previous studies. The max hop M was set to 3 for most datasets (Table II). However, for the E. coli dataset, it was reduced to 2 when employing one-dimensional homology due to memory constraints. Conversely, for the Power dataset, the max hop was set to 7 because it does not demand heavy memory and computation time. The sigmoid function was employed for the activation function of the PHLP classifier. Tables III and IV present the results of the hybrid methods using SEAL [24] and WP [28], respectively. For these experiments, a two-layer MLP was used for the MLP Φ in Step 3 of Section III-H. We set the k-hops following the original methods, SEAL and WP, and the max hops M of MA-PHLP were set as the k, except for the Power dataset. For the Power dataset, we set the k-hop to 1-hop and max hop M to 7, respectively, which is discussed in detail in Section IV-D.

B. Results

Results of MA-PHLP. Table II presents the AUC scores for each model on the benchmark datasets. Bold marks the best results, and underline indicates the second-best results. The results of AA, Katz, WLK, WLNM, N2V, SPC, MF, LINE, and SEAL are copied from SEAL [24] for comparison. The MA-PHLP demonstrates high performance across most datasets, achieving competitive scores. The proposed model outperforms several baselines, falling between the SEAL and WP models in terms of the AUC score. Notably, for the Power dataset, MA-PHLP achieves the highest AUC score, indicating its effectiveness in capturing link patterns.

Results of Hybrid Methods. Simply concatenating the PI

LINK PREDICTION PERFORMANCE MEASURED BY THE AUC ON BENCHMARK DATASETS (90% OBSERVED LINKS)

Dataset	USAir	NS	PB	Yeast	C. ele	Power	Router	E. coli
AA	95.06 ± 1.03	94.45 ± 0.93	92.36 ± 0.34	89.43 ± 0.62	86.95 ± 1.40	58.79 ± 0.88	56.43 ± 0.51	95.36 ± 0.34
Katz	92.88 ± 1.42	94.85 ± 1.10	92.92 ± 0.35	92.24 ± 0.61	86.34 ± 1.89	65.39 ± 1.59	38.62 ± 1.35	93.50 ± 0.44
PR	94.67 ± 1.08	94.89 ± 1.08	93.54 ± 0.41	92.76 ± 0.55	90.32 ± 1.49	66.00 ± 1.59	38.76 ± 1.39	95.57 ± 0.44
WLK	96.63 ± 0.73	98.57 ± 0.51	93.83 ± 0.59	95.86 ± 0.54	89.72 ± 1.67	82.41 ± 3.43	87.42 ± 2.08	96.94 ± 0.29
WLNM	95.95 ± 1.10	98.61 ± 0.49	93.49 ± 0.47	95.62 ± 0.52	86.18 ± 1.72	84.76 ± 0.98	94.41 ± 0.88	97.21 ± 0.27
N2V	91.44 ± 1.78	91.52 ± 1.28	85.79 ± 0.78	93.67 ± 0.46	84.11 ± 1.27	76.22 ± 0.92	65.46 ± 0.86	90.82 ± 1.49
SPC	74.22 ± 3.11	89.94 ± 2.39	83.96 ± 0.86	93.25 ± 0.40	51.90 ± 2.57	91.78 ± 0.61	68.79 ± 2.42	94.92 ± 0.32
MF	94.08 ± 0.80	74.55 ± 4.34	94.30 ± 0.53	90.28 ± 0.69	85.90 ± 1.74	50.63 ± 1.10	78.03 ± 1.63	93.76 ± 0.56
LINE	81.47 ± 10.71	80.63 ± 1.90	76.95 ± 2.76	87.45 ± 3.33	69.21 ± 3.14	55.63 ± 1.47	67.15 ± 2.10	82.38 ± 2.19
SEAL	97.10 ± 0.87	98.25 ± 0.61	95.07 ± 0.39	97.60 ± 0.33	89.54 ± 1.23	86.21 ± 2.89	95.07 ± 1.63	97.57 ± 0.30
WP	$\mathbf{98.20 \pm 0.57}$	$\mathbf{99.12 \pm 0.45}$	$\mathbf{95.42 \pm 0.25}$	$\mathbf{98.21 \pm 0.17}$	$\mathbf{93.30 \pm 0.91}$	92.11 ± 0.76	$\mathbf{97.15 \pm 0.29}$	$\mathbf{98.54 \pm 0.19}$
MA-PHLP	97.10 ± 0.69	98.88 ± 0.45	95.10 ± 0.26	97.98 ± 0.22	90.33 ± 1.16	93.05 ± 0.45	96.30 ± 0.43	97.64 ± 0.20
MA-PHLP (dim0)	97.10 ± 0.73	98.78 ± 0.65	95.06 ± 0.28	97.98 ± 0.23	89.88 ± 1.22	$\mathbf{93.37 \pm 0.41}$	96.37 ± 0.43	97.72 ± 0.17

TABLE III
AUC SCORES FOR SEAL WITH AND WITHOUT TDA FEATURES

Dataset	SEAL	MA-PHLP + SEAL
USAir	97.10 ± 0.87	$\mathbf{97.41 \pm 0.62}$
NS	98.25 ± 0.61	$\mathbf{98.97 \pm 0.30}$
PB	95.07 ± 0.39	$\mathbf{95.14 \pm 0.39}$
Yeast	97.60 ± 0.33	$\mathbf{97.93 \pm 0.18}$
C.ele	89.54 ± 1.23	$\mathbf{89.61 \pm 1.12}$
Power	86.21 ± 2.89	$\mathbf{95.53 \pm 0.33}$
Router	95.07 ± 1.63	$\mathbf{96.15 \pm 1.26}$
E.coli	97.57 ± 0.30	$\mathbf{97.93 \pm 0.34}$

vector calculated using PHLP with the final output of the SEAL model increases AUC scores for all datasets, as listed in Table III. This outcome suggests that when the SEAL model lacks topological information for inference, the vectors calculated using PHLP can serve as additional inputs.

TABLE IV
AUC SCORES FOR WALKPOOL (WP)
WITH AND WITHOUT TDA FEATURES

Dataset	WP	MA-PHLP + WP
USAir	98.20 ± 0.57	$\mathbf{98.27 \pm 0.53}$
NS	99.12 ± 0.45	$\mathbf{99.24 \pm 0.32}$
PB	95.42 ± 0.25	$\mathbf{95.58 \pm 0.32}$
Yeast	98.21 ± 0.17	$\mathbf{98.25 \pm 0.18}$
C.ele	93.30 ± 0.91	$\mathbf{93.32 \pm 0.71}$
Power	92.11 ± 0.76	$\mathbf{96.09 \pm 0.38}$
Router	97.15 ± 0.29	$\mathbf{97.18 \pm 0.24}$
E.coli	98.54 ± 0.19	$\mathbf{98.57 \pm 0.20}$

Similarly, we attempted to hybridize PHLP with the current SOTA model, WP. As presented in Table IV, a slight increase in AUC scores is observed for all datasets. The Power dataset demonstrates significant improvement.

C. Ablation Study

Effects of Degree DRNL. To assess the proposed Degree DRNL regarding the influence of incorporating degree information on model performance, we conducted experiments using DRNL and Degree DRNL and compared the results. We used MA-PHLP (dim0) for the experiments. Table V presents the AUC scores of MA-PHLP (dim0) with DRNL and Degree DRNL. Across all datasets, MA-PHLP (dim0) yields higher

TABLE V
AUC SCORES FOR MA-PHLP (DIM0) BY NODE LABELING

Dataset	DRNL	Degree DRNL
USAir	96.73 ± 0.64	$\mathbf{97.10 \pm 0.73}$
NS	98.35 ± 0.58	$\mathbf{98.78 \pm 0.65}$
PB	94.49 ± 0.27	$\mathbf{95.06 \pm 0.28}$
Yeast	97.42 ± 0.27	$\mathbf{97.98 \pm 0.23}$
C.ele	88.97 ± 1.37	$\mathbf{89.88 \pm 1.22}$
Power	88.51 ± 0.81	$\mathbf{92.77 \pm 0.47}$
Router	96.21 ± 0.53	$\mathbf{96.37 \pm 0.43}$
E.coli	97.15 ± 0.18	$\mathbf{97.72 \pm 0.17}$

AUC scores when used with Degree DRNL than with DRNL. The substantial improvement observed in the Power dataset is noteworthy, where Degree DRNL yields an increase of over 4 points in the AUC score. These experiments demonstrate the importance of incorporating degree information into node labeling, revealing its efficacy in enhancing the performance of MA-PHLP.

TABLE VI
AUC SCORES FOR MA-PHLP (DIM0) WITH VARIOUS (k, l)-ANGLE HOPS

Dataset	(1,0)	(1,1)
USAir	$\mathbf{96.15 \pm 0.83}$	95.87 ± 0.83
NS	98.28 ± 0.55	$\mathbf{98.66 \pm 0.66}$
PB	93.95 ± 0.34	$\mathbf{94.46 \pm 0.36}$
Yeast	95.52 ± 0.32	$\mathbf{97.31 \pm 0.20}$
C.ele	86.18 ± 2.12	$\mathbf{87.57 \pm 1.20}$
Power	73.39 ± 0.99	$\mathbf{77.83 \pm 1.44}$
Router	92.09 ± 0.57	$\mathbf{93.25 \pm 0.47}$
E.coli	96.94 ± 0.24	$\mathbf{96.95 \pm 0.28}$

Dataset	(2,0)	(2,1)	(2,2)
USAir	96.69 ± 0.92	96.74 ± 0.84	$\mathbf{96.85 \pm 0.83}$
NS	$\mathbf{98.72 \pm 0.51}$	98.59 ± 0.65	98.56 ± 0.47
PB	94.78 ± 0.30	94.73 ± 0.30	$\mathbf{94.82 \pm 0.24}$
Yeast	$\mathbf{97.71 \pm 0.18}$	97.66 ± 0.27	97.58 ± 0.28
C.ele	88.86 ± 1.48	$\mathbf{89.16 \pm 1.31}$	89.08 ± 1.07
Power	80.27 ± 1.07	83.90 ± 1.29	$\mathbf{86.12 \pm 0.86}$
Router	95.65 ± 0.44	$\mathbf{95.71 \pm 0.39}$	94.51 ± 0.69
E.coli	97.26 ± 0.16	97.29 ± 0.24	$\mathbf{97.41 \pm 0.21}$

Angles of PHLP. Table VI presents the performance of PHLP (dim 0) concerning various (k, l)-angle hop subgraphs. Section III-B proposed angle hop subgraphs as an alternative to traditional k-hop subgraphs to capture information from

various perspectives. Moreover, MA-PHLP is proposed to aggregate information from multiple angles. To investigate performance when extracting information from specific angles, we conducted experiments using PHLP at different angles. We used only zero-dimensional PIs for the experiments. Overall, the results demonstrate that the performance is favorable for cases corresponding to the k-hop subgraph (where k and l are the same). However, some datasets perform better when k and l differ, highlighting the importance of varying angles to achieve the best performance. Therefore, using MA-PHLP is recommended to maximize performance consistently across datasets.

Comparison with TLC-GNN. To demonstrate that the proposed method extracts superior topological information compared to the conventional TLC-GNN approach, we conducted the same experiments. The TLC-GNN was constructed by augmenting the graph convolutional network (GCN) model with PI information. We replaced the PI component of the TLC-GNN model with the PI vector produced by MA-PHLP, resulting in the MA-PHLP-GNN. The zero-dimensional PH was employed in this study for fair comparison because TLC-GNN used only zero-dimensional PH. Additionally, we conducted experiments where the PH vectors were replaced with zero vectors, denoted as GCN. Table VII presents the experimental results.

TABLE VII
COMPARISON OF AUC SCORES WITH TLC-GNN

Dataset	GCN	TLC-GNN	MA-PHLP-GNN
Cora	92.20 ± 0.83	$\mathbf{93.16 \pm 0.56}$	93.14 ± 0.93
CiteSeer	86.52 ± 1.29	87.38 ± 0.97	$\mathbf{92.08 \pm 0.53}$
PubMed	96.63 ± 0.15	96.30 ± 0.25	$\mathbf{98.07 \pm 0.07}$

The TLC-GNN is employed when the given data includes node attributes. Hence, we conducted experiments using the following widely used benchmark datasets with node attributes: Cora [60], CiteSeer [61], and PubMed [62]. The MA-PHLP-GNN outperformed the TLC-GNN significantly on the CiteSeer and PubMed datasets while achieving similar performance on the Cora dataset. The TLC-GNN does not exhibit performance improvement for the PubMed dataset despite adding topological information. However, the proposed MA-PHLP-GNN demonstrates substantial performance enhancement. Although the proposed model is developed for datasets without node attributes, it exhibits effective performance on datasets with node attributes through hybridization with the existing methods: SEAL+PHLP, WP+PHLP, and MA-PHLP-GNN. These experiments verify the versatility and effectiveness of this approach across diverse datasets.

D. The hops and max hops of the hybrid methods

Determining the hyperparameters such as "hop" and "max hop" is crucial for the performance of the hybrid method. We conducted experiments to explore the effects of different combinations of these parameters. Given that the hybrid methods (e.g., MA-PHLP + SEAL and MA-PHLP + WP) exhibited the highest performance improvement on the Power dataset,

we conducted experiments on the Power dataset. Table VIII presents the AUC scores for varying hop (SEAL or WP) and max hop (MA-PHLP). For each target node, while the SEAL and WP extract a k-hop subgraph, the MA-PHLP calculates the PIs based on a subgraph with max hop M. When the parameter M is 1 or 2, the AUC scores are not robust to k, showing large variations; however, when M is 3, although MA-PHLP + SEAL still exhibits variations up to 2, MA-PHLP + WP shows only minor variations. As M exceeds 3, the AUC scores of MA-PHLP + SEAL and MA-PHLP + WP are robust to k, exhibiting little sensitivity (maximum 0.84) to variations. This suggests that setting both the hop and the max hop to identical values may be permissible without further searching for optimal hyperparameters.

V. ANALYSIS

A. Analysis of the PHLP

Figs. 6 and 7 visualize concatenated PIs to illustrate how MA-PHLP (dim0) extracts topological features for LP. We let $\mathcal{Z} \subset \mathbb{R}^{2 \times k \times r^2}$ be a set of vectors calculated by MA-PHLP, where k is the number of angles, and r denotes the PI resolution. For $(z_1, z_2) \in \mathcal{Z}$, $z_1 \in \mathbb{R}^{k \times r^2}$ is the concatenation of PIs for all angles with a target link, and $z_2 \in \mathbb{R}^{k \times r^2}$ is the concatenation for cases without a target link. We consider a function $h : \mathbb{R}^{k \times r^2} \to \mathbb{R}$ defined as $h(\vec{v}_1, ..., \vec{v}_k) = \frac{1}{k} \sum_{i=1}^{k} \|\vec{v}_i\|_1$, where $\vec{v}_i \in \mathbb{R}^{r^2}$ are PIs, and $\|\cdot\|_1$ denotes the L_1-norm. For visualization, we transform $\mathcal{Z}$ into points in $\mathbb{R}^2$ using the function G, defined as $G(z_1, z_2) = (h(z_1), h(z_2))$ for each $(z_1, z_2) \in \mathcal{Z}$.

We plot distributions of points separately for positive and negative links, considering both DRNL and Degree DRNL. The distributions of the NS and Yeast datasets between positive and negative links display significant differences, supporting the highest performance in Table V. In contrast, the distributions for the C. ele and Power datasets are the most similar when using Degree DRNL, correlating with the lowest scores in Table V.

B. Analysis of the Power Dataset

In most LP models, including the SOTA models SEAL and WP, the Power dataset tends to have the lowest AUC scores among the datasets. In Table II, the Power dataset is at the bottom in terms of scores across models (e.g., WLK, WLNM, MF, LINE, SEAL, and WP). However, the proposed model achieves the highest AUC scores on the Power dataset among baseline models, prompting an analysis of the reasons for this performance.

In Fig. 7, for DRNL, the Power dataset exhibits horizontal lines, indicating that the values $h(z_2)$ have a limited range of outcomes for vectors z_2 in cases without the target link; thus, the set of values $h(z_2)$ with the same value should be spread out. This observation implies that, for numerous subgraphs the calculation of PIs yields similar outcomes despite the differences in their topological structures, posing a challenge in distinguishing between them.

To address this problem, we applied Degree DRNL, which incorporates degree information. The points in Fig. 7 are

DRNL

Degree DRNL

Positive links   Negative links   Positive links   Negative links
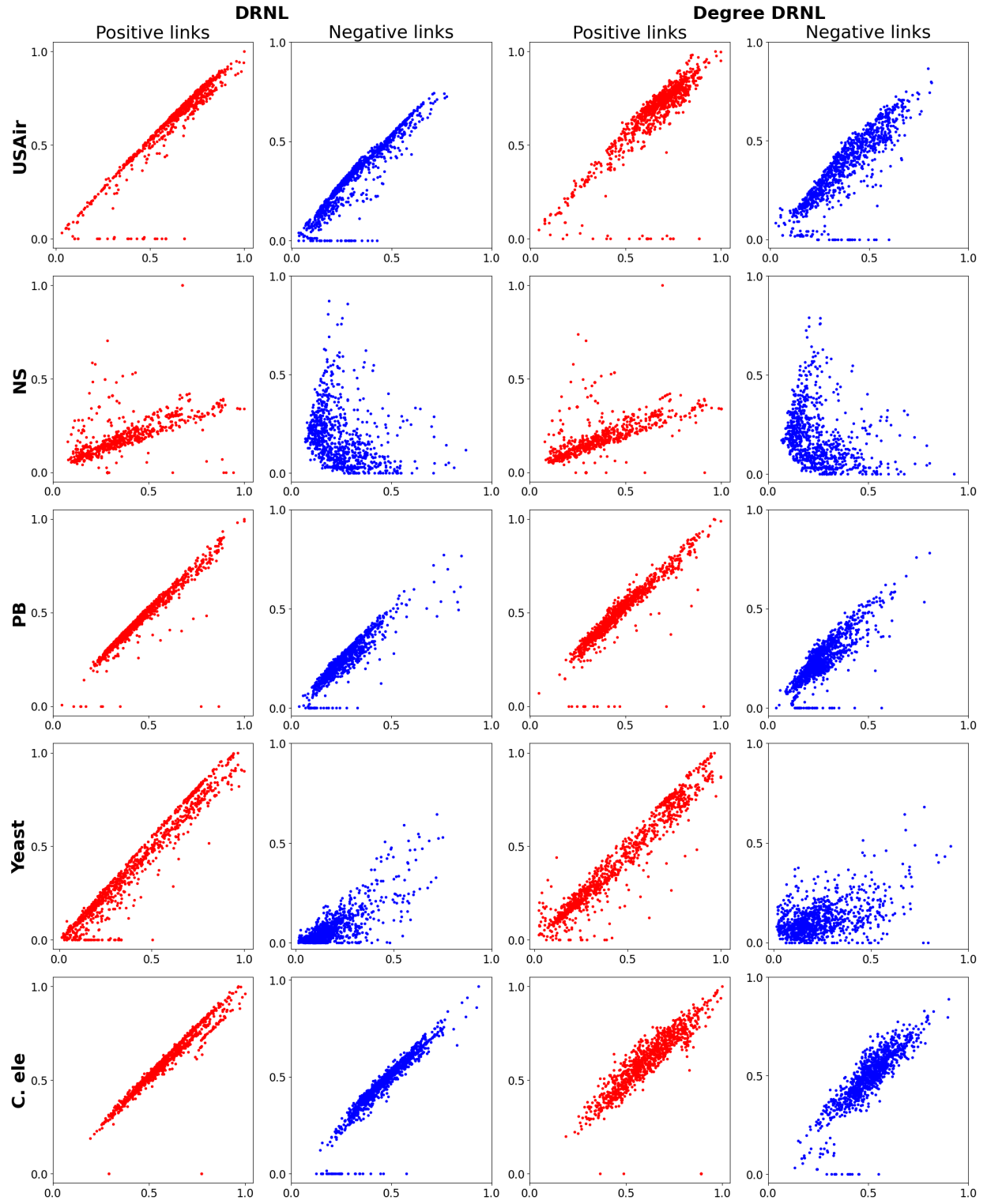
USAir

NS

PB

Yeast

C. ele

Fig. 6. Visualization of vectors calculated using MA-PHLP (dim0). For each dataset, the first and second columns depict the projections of persistence images (PIs) when double radius node labeling (DRNL) is applied for node labeling, and the third and fourth columns represent the values obtained when Degree DRNL is applied. The first and third columns plot the values produced from positive edges (i.e., target nodes labeled 1), and the second and fourth columns plot the values produced from negative edges (i.e., target nodes labeled 0).

TABLE VIII
AUC SCORES ON THE POWER DATASET VARYING k-HOP AND MAX HOP M OF THE HYBRID METHODS

	MA-PHLP (with max hop M)						
M	1	2	3	4	5	6	7
k	not robust to k			robust to k			
1	86.66 ± 0.56	90.22 ± 0.79	92.63 ± 0.54	94.50 ± 0.41	95.12 ± 0.40	95.46 ± 0.38	$\mathbf{95.53 \pm 0.33}$
2	91.40 ± 0.88	90.20 ± 0.80	92.50 ± 0.59	94.39 ± 0.39	95.00 ± 0.46	95.31 ± 0.40	$\mathbf{95.39 \pm 0.36}$
3	93.21 ± 0.64	92.79 ± 0.60	92.57 ± 0.58	94.22 ± 0.43	94.86 ± 0.42	$\mathbf{95.21 \pm 0.45}$	95.19 ± 0.44
4	94.51 ± 0.58	94.23 ± 0.34	94.21 ± 0.41	94.31 ± 0.40	94.80 ± 0.37	95.10 ± 0.33	$\mathbf{95.27 \pm 0.36}$
5	94.73 ± 0.56	94.45 ± 0.44	94.61 ± 0.51	94.80 ± 0.53	94.91 ± 0.54	95.13 ± 0.51	$\mathbf{95.19 \pm 0.46}$
6	94.58 ± 0.94	94.81 ± 0.32	94.87 ± 0.42	95.06 ± 0.50	95.11 ± 0.46	$\mathbf{95.25 \pm 0.45}$	95.25 ± 0.46
7	93.97 ± 0.73	94.22 ± 0.35	94.43 ± 0.44	94.78 ± 0.45	94.92 ± 0.39	$\mathbf{94.99 \pm 0.52}$	94.98 ± 0.39
k	not robust to k			robust to k			
1	87.53 ± 0.73	91.48 ± 0.64	93.55 ± 0.48	94.84 ± 0.43	95.53 ± 0.46	95.88 ± 0.31	$\mathbf{96.09 \pm 0.38}$
2	92.51 ± 0.58	91.59 ± 0.77	93.49 ± 0.58	94.83 ± 0.53	95.56 ± 0.59	95.88 ± 0.38	$\mathbf{96.06 \pm 0.45}$
3	94.04 ± 0.46	93.07 ± 0.67	93.61 ± 0.52	94.86 ± 0.54	95.61 ± 0.60	95.86 ± 0.40	$\mathbf{96.00 \pm 0.52}$
4	93.55 ± 0.71	92.61 ± 0.76	93.68 ± 0.55	94.85 ± 0.55	95.59 ± 0.58	95.87 ± 0.38	$\mathbf{96.03 \pm 0.45}$
5	93.40 ± 0.70	92.64 ± 0.69	93.66 ± 0.53	94.84 ± 0.54	95.55 ± 0.59	95.85 ± 0.39	$\mathbf{96.04 \pm 0.52}$
6	93.34 ± 0.75	92.66 ± 0.72	93.64 ± 0.55	94.91 ± 0.57	95.55 ± 0.58	95.85 ± 0.44	$\mathbf{95.98 \pm 0.55}$
7	93.30 ± 0.73	92.61 ± 0.69	93.65 ± 0.56	94.87 ± 0.56	95.56 ± 0.58	95.90 ± 0.39	$\mathbf{96.01 \pm 0.52}$
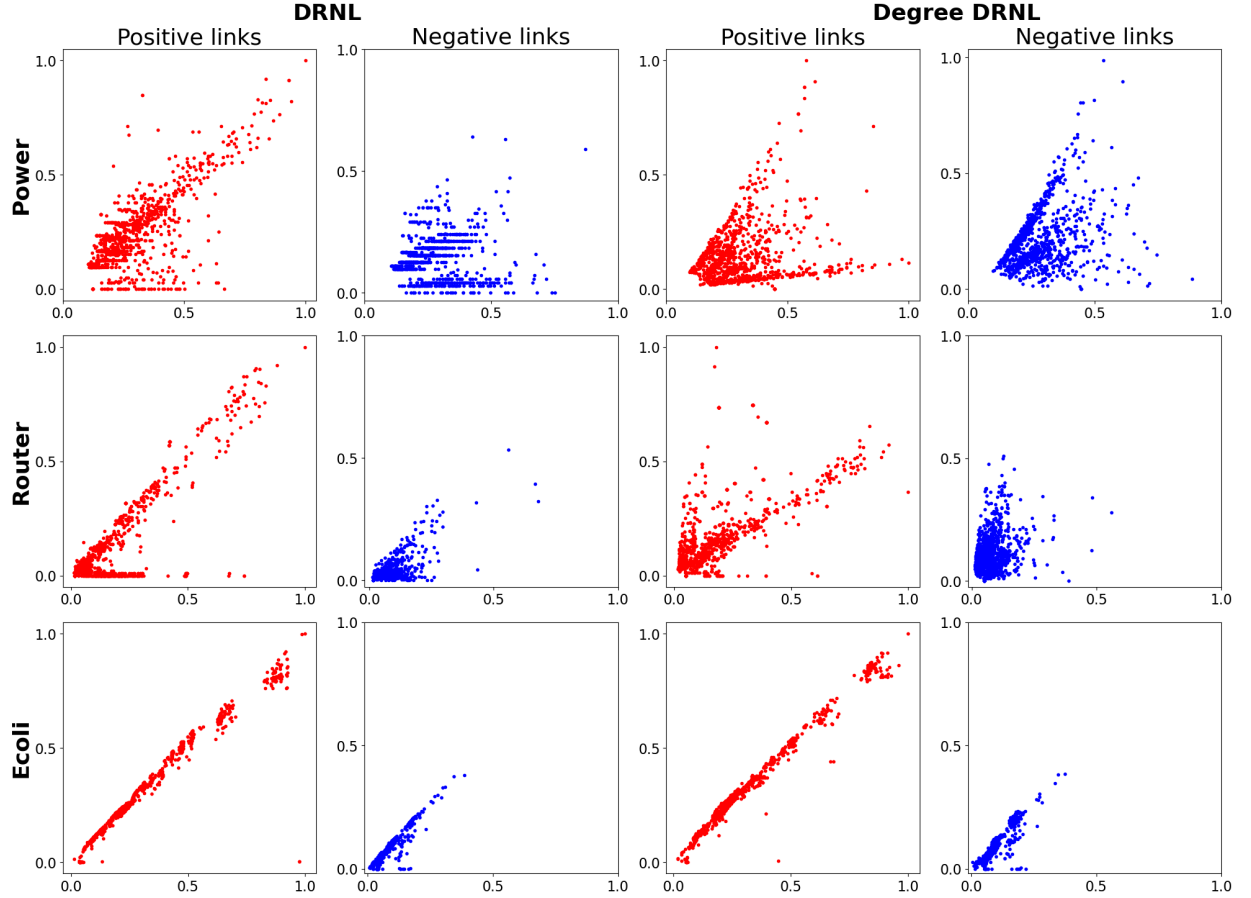
Fig. 7. Visualization of vectors calculated using MA-PHLP (dim0).

distributed without horizontal lines, leading to the highest score increase, as listed in Table V.

The performance of heuristic methods, such as AA, Katz, and PR, tend to be similar to random guessing on datasets with low density, particularly in the cases of the Power and Router datasets. Embedding methods also display low performance. In contrast, the GNN-based methods demonstrate improved performance using subgraphs and the network learning ability. However, the performance for the Power dataset is significantly lower than that for the Router dataset.

TABLE IX
AVERAGE NUMBER OF NODES IN SUBGRAPHS FOR THE POWER AND ROUTER DATASETS

	Power		Router	
	positive	negative	positive	negative
1-hop	8.03	9.12	5.11	6.72
2-hop	22.26	24.85	29.21	13.94
3-hop	43.11	49.50	120.35	55.22
4-hop	71.72	82.16	411.87	176.34
5-hop	99.28	116.75	740.80	411.35
6-hop	136.23	158.27	1272.42	852.13
7-hop	182.22	210.35	1835.46	1498.58

To bridge this gap, we analyzed subgraphs with node labeling. The number of nodes within the selected subgraphs between positive and negative links was significantly different on the Router dataset but not the Power dataset (Table IX). This difference is attributed to the presence of the hub nodes in the Router dataset, which are connected to numerous nodes. Thus, the subgraphs corresponding to positive links tend to have more nodes than those corresponding to negative links.

TABLE X
COMPARISON OF MODELS BY MAX HOP SETTINGS ON THE POWER AND ROUTER DATASETS

	Model	MA-PHLP	MA-PHLP	WP	MA-PHLP + WP
	Center	target	random	-	random
Power	1-hop	78.05 ± 1.20	85.66 ± 0.86	80.24 ± 0.95	87.53 ± 0.73
	2-hop	86.34 ± 1.04	90.52 ± 0.73	89.40 ± 1.00	91.59 ± 0.77
	3-hop	89.65 ± 0.64	91.90 ± 0.58	$\mathbf{92.11 \pm 0.77}$	93.61 ± 0.52
	4-hop	91.38 ± 0.53	92.67 ± 0.55	91.67 ± 0.80	94.85 ± 0.55
	5-hop	92.27 ± 0.40	93.06 ± 0.44	91.39 ± 0.78	95.55 ± 0.59
	6-hop	92.77 ± 0.47	93.16 ± 0.49	91.55 ± 0.83	95.85 ± 0.44
	7-hop	$\mathbf{93.06 \pm 0.43}$	$\mathbf{93.37 \pm 0.41}$	91.50 ± 0.89	$\mathbf{96.01 \pm 0.52}$
Router	1-hop	93.12 ± 0.45	93.40 ± 0.46	94.48 ± 0.36	94.83 ± 0.41
	2-hop	95.96 ± 0.40	95.70 ± 0.45	97.15 ± 0.27	97.22 ± 0.23
	3-hop	96.38 ± 0.41	96.11 ± 0.43	$\mathbf{97.28 \pm 0.24}$	$\mathbf{97.42 \pm 0.27}$
	4-hop	96.45 ± 0.40	96.22 ± 0.43	OOM[1]	OOM
	5-hop	$\mathbf{96.46 \pm 0.42}$	$\mathbf{96.24 \pm 0.48}$	OOM	OOM
	6-hop	96.44 ± 0.45	96.23 ± 0.47	OOM	OOM
	7-hop	96.43 ± 0.45	96.19 ± 0.49	OOM	OOM

However, the Power dataset does not have hub nodes, and the number of nodes in the subgraph of positive links remains small. We randomly changed the center nodes (a, b) for node labeling $f_{\mathrm{degdrnl}}^{(a,b)}$ increasing the performance, as listed in Table X. This outcome highlights that setting target nodes as the center nodes may not effectively analyze the topological structure in the case of small graphs. Furthermore, the performance for the Power dataset continues to increase with increasing hops (Table X), achieving an AUC score of 95.87, which is significantly better than that of 92.11 for WP.

VI. CONCLUSION

This paper proposes PHLP, an explainable method that applies PH to analyze the topological structure of graphs to overcome the limitations of GNN-based methods for LP. By employing the proposed methods, such as angle hop subgraphs and Degree DRNL, PHLP improves the analysis of the topological structure of graphs. The experimental results

[1]OOM denotes "out of GPU memory".

demonstrate that the proposed PHLP method achieves competitive performance across benchmark datasets, even SOTA performance, especially on the Power dataset. Additionally, when integrated with existing GNN-based methods, PHLP improves performance across all datasets. By analyzing the topological information of the given graphs, PHLP addresses the limitations of GNN-based methods and enhances overall performance. As demonstrated, PHLP provides explainable algorithms without relying on complex deep learning techniques, providing insight into the factors that significantly influence performance for the LP problem of graph data.

REFERENCES

[1] Z. Zhang, P. Cui, and W. Zhu, "Deep learning on graphs: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 249–270, 2020.

[2] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 4–24, 2020.

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

[4] L. Yao, L. Wang, L. Pan, and K. Yao, "Link prediction based on common-neighbors for dynamic social network," *Procedia Computer Science*, vol. 83, pp. 82–89, 2016.

[5] M. Fire, L. Tenenboim, O. Lesser, R. Puzis, L. Rokach, and Y. Elovici, "Link prediction in social networks using computationally efficient topological features," in *2011 IEEE third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computing*. IEEE, 2011, pp. 73–80.

[6] S. M. Kazemi and D. Poole, "Simple embedding for link prediction in knowledge graphs," *Advances in neural information processing systems*, vol. 31, 2018.

[7] M. Nayyeri, G. M. Cil, S. Vahdati, F. Osborne, A. Kravchenko, S. Angioni, A. Salatino, D. R. Recupero, E. Motta, and J. Lehmann, "Link prediction of weighted triples for knowledge graph completion within the scholarly domain," *Ieee Access*, vol. 9, pp. 116 002–116 014, 2021.

[8] Z. Stanfield, M. Coşkun, and M. Koyutürk, "Drug response prediction as a link prediction problem," *Scientific reports*, vol. 7, no. 1, p. 40321, 2017.

[9] E. Nasiri, K. Berahmand, M. Rostami, and M. Dabiri, "A novel link prediction algorithm for protein-protein interaction networks by attributed graph embedding," *Computers in Biology and Medicine*, vol. 137, p. 104772, 2021.

[10] C. Lei and J. Ruan, "A novel link prediction algorithm for reconstructing protein–protein interaction networks by topological similarity," *Bioinformatics*, vol. 29, no. 3, pp. 355–364, 2013.

[11] I. A. Kovács, K. Luck, K. Spirohn, Y. Wang, C. Pollis, S. Schlabach, W. Bian, D.-K. Kim, N. Kishore, T. Hao *et al.*, "Network-based prediction of protein interactions," *Nature communications*, vol. 10, no. 1, p. 1240, 2019.

[12] N. Brockmann, E. Elson Kosasih, and A. Brintrup, "Supply chain link prediction on uncertain knowledge graph," *ACM SIGKDD Explorations Newsletter*, vol. 24, no. 2, pp. 124–130, 2022.

[13] A. Brintrup, P. Wichmann, P. Woodall, D. McFarlane, E. Nicks, and W. Krechel, "Predicting hidden links in supply networks," *Complexity*, vol. 2018, pp. 1–12, 2018.

[14] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A: statistical mechanics and its applications*, vol. 390, no. 6, pp. 1150–1170, 2011.

[15] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *science*, vol. 286, no. 5439, pp. 509–512, 1999.

[16] T. Zhou, L. Lü, and Y.-C. Zhang, "Predicting missing links via local information," *The European Physical Journal B*, vol. 71, pp. 623–630, 2009.

[17] S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual web search engine," *Computer networks*, vol. 56, no. 18, pp. 3825–3833, 2012.

[18] G. Jeh and J. Widom, "Simrank: a measure of structural-context similarity," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002, pp. 538–543.

[19] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[20] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.

[21] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[22] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th international conference on world wide web*, 2015, pp. 1067–1077.

[23] T. N. Kipf and M. Welling, "Variational graph auto-encoders," *arXiv preprint arXiv:1611.07308*, 2016.

[24] M. Zhang and Y. Chen, "Link prediction based on graph neural networks," *Advances in neural information processing systems*, vol. 31, 2018.

[25] S. Yun, S. Kim, J. Lee, J. Kang, and H. J. Kim, "Neo-gnns: Neighborhood overlap-aware graph neural networks for link prediction," *Advances in Neural Information Processing Systems*, vol. 34, pp. 13 683–13 694, 2021.

[26] C. Mavromatis and G. Karypis, "Graph infoclust: Leveraging cluster-level node information for unsupervised graph representation learning," *arXiv preprint arXiv:2009.06946*, 2020.

[27] Z. Yan, T. Ma, L. Gao, Z. Tang, and C. Chen, "Link prediction with persistent homology: An interactive view," in *International conference on machine learning*. PMLR, 2021, pp. 11 659–11 669.

[28] L. Pan, C. Shi, and I. Dokmanić, "Neural link prediction with walk pooling," *arXiv preprint arXiv:2110.04375*, 2021.

[29] S. Huber, "Persistent homology in data science," in *Data Science–Analytics and Applications: Proceedings of the 3rd International Data Science Conference–iDSC2020*. Springer, 2021, pp. 81–88.

[30] T. K. Dey and Y. Wang, *Computational topology for data analysis*. Cambridge University Press, 2022.

[31] M. Horn, E. De Brouwer, M. Moor, Y. Moreau, B. Rieck, and K. Borgwardt, "Topological graph neural networks," *arXiv preprint arXiv:2102.07835*, 2021.

[32] X. Ye, F. Sun, and S. Xiang, "Treph: A plug-in topological layer for graph neural networks," *Entropy*, vol. 25, no. 2, p. 331, 2023.

[33] M. Carrière, F. Chazal, Y. Ike, T. Lacombe, M. Royer, and Y. Umeda, "Perslay: A neural network layer for persistence diagrams and new graph topological signatures," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2786–2796.

[34] F. M. Taiwo, U. Islambekov, and C. G. Akcora, "Explaining the power of topological data analysis in graph machine learning," *arXiv preprint arXiv:2401.04250*, 2024.

[35] T. Wen, E. Chen, and Y. Chen, "Tensor-view topological graph neural network," *arXiv preprint arXiv:2401.12007*, 2024.

[36] J. Immonen, A. Souza, and V. Garg, "Going beyond persistent homology using persistent homology," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[37] C. Ying, X. Zhao, and T. Yu, "Boosting graph pooling with persistent homology," *arXiv preprint arXiv:2402.16346*, 2024.

[38] Q. Zhao and Y. Wang, "Learning metrics for persistence-based summaries and applications for graph classification," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[39] Y. Chen, B. Coskunuzer, and Y. Gel, "Topological relational learning on graphs," *Advances in neural information processing systems*, vol. 34, pp. 27 029–27 042, 2021.

[40] Q. Zhao, Z. Ye, C. Chen, and Y. Wang, "Persistence enhanced graph neural network," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2896–2906.

[41] M. Nickel, X. Jiang, and V. Tresp, "Reducing the rank in relational factorization models by including observable patterns," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[42] L. F. Ribeiro, P. H. Saverese, and D. R. Figueiredo, "struc2vec: Learning node representations from structural identity," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 2017, pp. 385–394.

[43] L. Vietoris, "Über den höheren zusammenhang kompakter räume und eine klasse von zusammenhangstreuen abbildungen," *Mathematische Annalen*, vol. 97, no. 1, pp. 454–472, 1927.

[44] M. Gromov, "Hyperbolic groups," in *Essays in group theory*. Springer, 1987, pp. 75–263.

[45] Edelsbrunner, Letscher, and Zomorodian, "Topological persistence and simplification," *Discrete & computational geometry*, vol. 28, pp. 511–533, 2002.

[46] H. Adams, T. Emerson, M. Kirby, R. Neville, C. Peterson, P. Shipman, S. Chepushtanova, E. Hanson, F. Motta, and L. Ziegelmeier, "Persistence images: A stable vector representation of persistent homology," *Journal of Machine Learning Research*, vol. 18, 2017.

[47] A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," *Pattern recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.

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

[49] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, vol. 30, no. 1-7, pp. 107–117, 1998.

[50] N. Shervashidze, P. Schweitzer, E. J. Van Leeuwen, K. Mehlhorn, and K. M. Borgwardt, "Weisfeiler-lehman graph kernels." *Journal of Machine Learning Research*, vol. 12, no. 9, 2011.

[51] M. Zhang and Y. Chen, "Weisfeiler-lehman neural machine for link prediction," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 2017, pp. 575–583.

[52] L. Tang and H. Liu, "Leveraging social media networks for classification," *Data Mining and Knowledge Discovery*, vol. 23, pp. 447–478, 2011.

[53] V. Batagelj and A. Mrvar, "Pajek datasets," http://vlado.fmf.uni-lj.si/pub/networks/data/, 2006.

[54] M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical review E*, vol. 74, no. 3, p. 036104, 2006.

[55] R. Ackland *et al.*, "Mapping the us political blogosphere: Are conservative bloggers more prominent?" in *BlogTalk Downunder 2005 Conference, Sydney*. BlogTalk Downunder 2005 Conference, Sydney, 2005.

[56] C. Von Mering, R. Krause, B. Snel, M. Cornell, S. G. Oliver, S. Fields, and P. Bork, "Comparative assessment of large-scale data sets of protein–protein interactions," *Nature*, vol. 417, no. 6887, pp. 399–403, 2002.

[57] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world'networks," *nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[58] N. Spring, R. Mahajan, and D. Wetherall, "Measuring isp topologies with rocketfuel," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 4, pp. 133–145, 2002.

[59] M. Zhang, Z. Cui, S. Jiang, and Y. Chen, "Beyond link prediction: Predicting hyperlinks in adjacency space," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[60] A. K. McCallum, K. Nigam, J. Rennie, and K. Seymore, "Automating the construction of internet portals with machine learning," *Information Retrieval*, vol. 3, pp. 127–163, 2000.

[61] C. L. Giles, K. D. Bollacker, and S. Lawrence, "Citeseer: An automatic citation indexing system," in *Proceedings of the third ACM conference on Digital libraries*, 1998, pp. 89–98.

[62] G. Namata, B. London, L. Getoor, B. Huang, and U. Edu, "Query-driven active surveying for collective classification," in *10th international workshop on mining and learning with graphs*, vol. 8, 2012, p. 1.