Disentangling Node Attributes from Graph Topology for Improved Generalizability in Link Prediction

Anonymous Author(s)

ABSTRACT

Predicting whether two nodes u and v are connected in a graph (a.k.a. link prediction) is useful in many application domains, from friend recommendations in social networks to drug discovery in drug-target interaction networks. The most common setting in link prediction is the *transductive* scenario, where both nodes u and v are observed during training. When one or both nodes are not observed during training, the link prediction setting is referred to as *semi-inductive* and *inductive*, respectively. In the absence of topological information, link prediction models are forced to use node attributes to make accurate predictions for previously unobserved nodes. For example, recommending a new item to a user in the semi-inductive setting or identifying the interaction between a new protein target and a newly developed drug in the inductive setting requires meaningful pairwise learning of node attributes. In addition, the semi-inductive and inductive scenarios are of interest for predicting connections in temporal networks, where newly arrived nodes connect to temporal instances of the graph, and create a topology that evolves over time.

We investigate the interplay between node attributes and graph topology from an information-theoretic standpoint and show how node attributes, which contain information beyond graph topology, improve the generalization power of link prediction models quantified in terms of inductive link prediction performance. UPNA (Unsupervised Pre-training of Node Attributes) on large corpora allows us to learn the latent mechanism of graph generation independent of the observed graph. By gaining insight into the graph growth mechanism via node attributes, we remove the observational bias associated with a fixed snapshot of the training graph and are able to make meaningful predictions about unobserved nodes while bypassing topological shortcuts. Pre-training of node attributes improves inductive link prediction performance by multiple folds compared to the state-of-the-art (between 3× to 34× on benchmark datasets). *UPNA* can be extended to any pairwise learning task, such as the cold-start problem in recommender systems and entity resolution. *UPNA* integrated with existing link prediction models, which leverage attribute information, can improve the generalizability of a wide class of state-of-the-art methods in link prediction.

CCS CONCEPTS

• **Computing methodologies** → **Learning latent representations**; **Network science**; • **Mathematics of computing** → **Graphs and surfaces**.

KEYWORDS

Graph machine learning, Link prediction, Inductive learning, Generalizability, Pre-training, Unsupervised learning

KDD '23 Research Track, August 06–10, 2023, Long Beach, CA, https://kdd.org/kdd2023/.

ACM Reference Format:

Anonymous Author(s). 2023. Disentangling Node Attributes from Graph Topology for Improved Generalizability in Link Prediction. In *KDD '23: 29th ACM SIGKDD Conference On Knowledge Discovery and Data Mining, August 06–10, 2019, Long Beach, CA.* ACM, New York, NY, USA, 11 pages.

1 INTRODUCTION

Graph datasets are ubiquitous. Examples include friendship networks [6], collaboration networks [76], protein interaction networks [61], power grids [58], and transportation networks [45]. Real-world graphs are often sparse and partially observed, which makes predicting the unobserved links a problem of great interest [43]. Applications of link prediction include predicting unknown protein interactions, exploring drug responses [68], recommending products to users [40], completing knowledge graphs [57], and suggesting friends in social networks [1].

Link prediction is a well-researched problem, and numerous methods have been developed, which include similarity-based indices, probabilistic methods, dimensionality reduction approaches, etc. [38]. Latent representations of nodes and edges, e.g., Node2Vec [25], are often used for link prediction [11, 60, 71, 75]. These embeddings encode the graph topology in low-dimensional feature vectors, which are then used for training on the observed links. Simple features representing the graph topology can also achieve similar performance as these deep learning-based latent features [23]. In a recent work, the importance of both the graph topology and the node attributes in link prediction has been emphasized [3].

However, the majority of link prediction approaches and their benchmarks focus on transductive link prediction, where the train and test graphs share the same set of nodes [15, 30, 77]. Bonner et al. [9] and Chatterjee et al. [12] discuss how transductive link prediction is driven mainly by the training graph topology and is biased by the degree distribution of the observed graph. Their observations were on biological networks. Similar observations have been made on another downstream task on graphs, namely, node classification, where label propagation achieves comparable or superior performance compared to state-of-the-art deep models [33]. Many real-world applications require making semi-inductive and inductive link predictions on newly observed nodes, which requires learning from the node attributes independent of the graph topology. The latent graph generation model devoid of the observational bias of the training graph instance is the key to making accurate semi-inductive and inductive link predictions. Cold-start is an extensively explored problem in recommendation systems, which requires suggesting a new product to an existing user [48]. This resembles a semi-inductive setting, where one node of the test link is seen during training, while the other one is a never-before-seen node. When both nodes are unseen, the prediction task is more challenging and resembles an inductive test scenario. For example, AI-Bind [12, 51] solves inductive link prediction in protein-ligand interaction networks, where both the protein and the ligand in the

test set are unseen during training. Inductive link prediction in knowledge graphs has recently gained significant attention [19].

Temporal networks demand learning beyond the limited aspects of static graphs while exploring dynamic complex systems [29]. Static network analysis often produces misleading insights into spreading processes and community structures in graphs [65]. Understanding the complex dynamic processes on networks requires unveiling the mechanism of evolution of the temporal graphs and the underlying causal topology. In a discrete-time system, the newly arrived nodes at each instance attach to the previous temporal snapshot of the graph, devising a semi-inductive or sometimes inductive link prediction scenario.

Identifying useful node attributes is necessary for improving the performance of node classification, link prediction, and graph data augmentation [13]. Deep adversarial learning and variational auto-encoder-based approaches have been proposed for creating real-world attributes in improving the downstream tasks. Yet, the literature lacks an understanding of the interplay between the graph topology and the node attributes and demands a formal quantification of the goodness of node attributes for adopting the best-suited attributes in improving and generalizing these downstream tasks.

Contributions: (1) We explore topological shortcuts and observe how the transductive link prediction performance is largely driven by the high-degree nodes, leading to an observational bias and poor downstream performance on the low-degree and unobserved nodes. **(2)** We demonstrate the importance of learning the latent graph generation mechanism from the node attributes for making accurate link predictions for the isolated newly arrived nodes. **(3)** We quantify the generalization power of the link prediction models in terms of the inductive test performance and establish a relation between the generalization power of the link prediction models and the information contained in the node attributes. **(4)** We propose UPNA (Unsupervised Pre-training of Node Attributes) on large corpora independent of the training graph to improve the link prediction generalizability. We validate our proposed method on both static and time-evolving graphs.

2 PROBLEM FORMULATION

Static Graphs

Let $G = (V, E, X)$ be a graph instance, where V is the set of vertices (or nodes), E is the set of edges (or links), and the X matrix contains the node attributes. Here we consider an undirected unipartite graph. This formulation can be extended to directed graphs, bipartite graphs, and multilayered graphs. For example, G may be a protein-protein interaction network, where the nodes are the proteins, the links are the interactions between the proteins, and the node attributes are the embeddings of the molecular structures of the proteins using ProtVec [4].

A link prediction model w may be constructed using supervised learning on the set of links E. We partition the edge set into observed and unobserved edges $E = E_o \cup E_u$. We learn a function that maps the observed nodes and the node attributes to the observed edges $\{V_o, X_o\} \rightarrow E_o$, and hope that it will generalize to the unobserved edges E_u. Furthermore, we define three types of link prediction scenarios based on the observed and the unobserved nodes V_o and V_u, respectively:

- Transductive: Predicting $(a, b) \in E_u$, where $a, b \in V_o$,
- Semi-inductive: Predicting $(a, b) \in E_u$, where $a \in V_o$ and $b \in V_u$ or vice-versa,
- Inductive: Predicting $(a, b) \in E_u$, where $a, b \in V_u$.

In this work, we explore the semi-inductive and inductive link prediction scenarios. The link prediction model takes input $\{V_o, X_o, E_o\}$, and makes predictions on E_u induced by V_u.

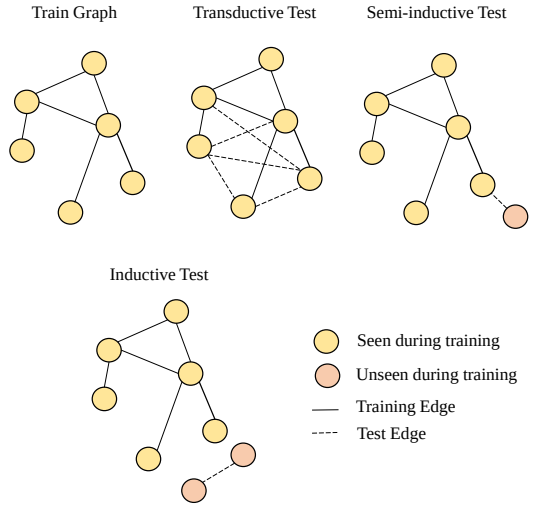

Figure 1: Transductive, semi-inductive, and inductive link prediction tasks in static graphs.

Temporal Graphs

We also consider discrete-time temporal graphs, i.e. we consider the temporal snapshots of the graph instances at the intervals of Δt, where the time stamps are t_0, $t_1 = t_0 + \Delta t$, ..., $t_n = t_0 + (n-1)\Delta t$.

$G_{t_0} = (V_{t_0}, E_{t_0}, X_{t_0}), G_{t_1} = (V_{t_1}, E_{t_1}, X_{t_1}), ..., G_{t_n} = (V_{t_n}, E_{t_n}, X_{t_n})$ are the temporal graph instances of a time-evolving network corresponding to the time stamps t_0, t_1, ..., t_n, respectively. The unobserved node set from time t_j to t_{j+1} is $V_u = V_{t_{j+1}} \setminus V_{t_j}$, and the unobserved edge set is represented by $E_u = E_{t_{j+1}} \setminus E_{t_j}$. Under this setting, the link prediction model takes input $G_{t_j} = (V_{t_j}, E_{t_j}, X_{t_j})$, and makes predictions on E_u induced by V_u.

3 TOPOLOGICAL SHORTCUTS

What is the influence of the observation bias on link prediction in static graphs? We show that the transductive link prediction performance of the state-of-the-art models is largely driven by the graph topology and can be achieved by much simpler non-machine-learning algorithms. We perform experiments on the Open Graph Benchmark (OGB) datasets [30] showing that the performance of the top deep models on the link prediction benchmark ogbl-ddi (drug-drug interaction network) can be replicated by simple configurations models, which ignore the node attributes and rely solely on the topology of the training graph. This is strong evidence that these deep models are reliant on shortcuts exploiting the graph

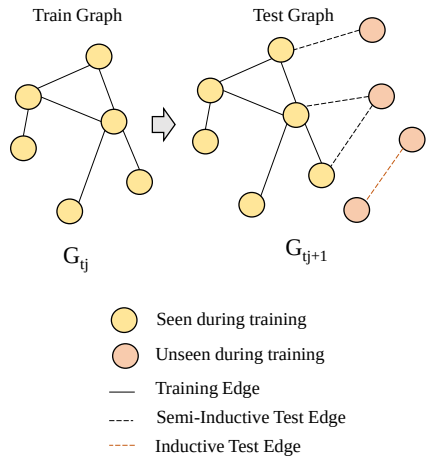

Figure 2: Semi-inductive and inductive link prediction task in discrete temporal graphs.

topology and ignoring the node attributes. This is problematic since the graph topology cannot be used for generalization to unseen nodes in semi-inductive and inductive tests.

Shortcuts in transductive learning have been investigated in multiple previous studies. Ghasemian et al. [23] showed that link prediction models stacking topological features have sub-optimal link prediction performance on graph datasets across different domains such as social networks, biological networks, information networks, and transportation networks. Configuration models use only the degree sequence of the training graph, and through an entropy maximization algorithm produce the probabilities associated with the unobserved links. Chatterjee et al. [12] demonstrated that a duplex bipartite configuration model, using only the degree sequences of the proteins and the ligands in the training drug-target interaction network, achieves test performance in predicting transductive protein-ligand interactions comparable to state-of-the-art deep neural networks [31, 32]. Similar observations have been made for block-approximated exponential random graph models [2, 49] and simple link prediction heuristics [17, 50] in link prediction.

Here, we consider two types of configuration models. The first one is a traditional soft configuration model [73], which takes as input the number of nodes n and the degree sequence of the training graph $\mathbf{k}$. This model depends only on the degree sequence of the observed positive edges and does not require any information on the negative edges. The traditional configuration model is represented by the Soft Configuration Model SCM$(n, \mathbf{k})$, where $\mathbf{k} = \{k_i\}_{i=1}^{n}$, $k_i \geq 0$ is the degree sequence of the nodes in training.

The SCM is an exponential random graph model [18, 52, 53] with the probability of observing a graph configuration $P(G) = \frac{B(G)}{Z}$, where $B(G) = \exp(-\sum_{i=1}^{n} \lambda_i d_i(G))$, the Lagrange multipliers $\{\lambda_i\}$ are such that $\langle d_i \rangle = \sum_{j=1}^{n} p_{ij} = k_i$ for all nodes $i \in \{1, 2, \ldots, n\}$, the Boltzmann factor $B = \sum_G B(G)$, $d_i(G) = \sum_{j=1}^{n} G_{ij}$ is the degree of the node i, and $\{G_{ij}\}_{i,j=1}^{n}$ is the adjacency matrix of the training graph G. By entropy maximization, we get the link probability between the nodes i and j as:

$$p_{ij} = \frac{1}{e^{\lambda_i + \lambda_j} + 1}, \tag{1}$$

We also use a duplex configuration model developed by Menichetti et al. [54], which takes into account both positive and negative edges for making the link predictions. We use the same negative sampling strategy as OGB, randomly sampling the negative edges from the unobserved link set in the training graph, and keeping the number of positive and negative edges the same in training. The duplex configuration model represents the positive and the negative training graphs as a duplex network. Thereafter, it runs an entropy maximization algorithm jointly on both layers.

Using the link probabilities in Eq. 1 for the traditional configuration model, and the formulation developed in [54] for the unipartite duplex configuration model, we compute the transductive test performances in terms of the OGB-recommended metric Hit@Top K on the benchmark ogbl-ddi dataset. In Table 1, we compare the performances of the traditional and the duplex configuration models with the three top-performing models, namely, Adaptive Graph Diffusion Networks (AGDN, [70]), Path-aware Siamese Graph Neural Network (PSG, [46]), and Pairwise Learning for Neural Link Prediction (PLNLP, [80]) in the transductive setting. We use the OGB-provided benchmark train-validation-tests split. We observe that the configuration models outperform the state-of-the-art neural network models. In the transductive link prediction setting, the fact that simple configuration models learn only from the topological data can outperform recent deep learning-based models trained on both the topology and the node attributes provides evidence that these deep learning models derive much of their predictive power from the topological features alone. Similar to our observation, a recent work [69] shows that classic graph structural features outperform graph embedding-based methods in another downstream task on graphs (namely, community labeling) when the performance measure is Hits@Top K. Similar observations on topological shortcuts are made on another benchmark Deep Graph Library (DGL) [78] here. Topological shortcuts are also reflected in other performance metrics such as Area Under the Receiver Operating Characteristics (AUROC) and Area Under the Precision Recall Curve (AUPRC) [12].

Table 1: The traditional and the duplex configuration models outperform the state-of-the-art neural network models in terms of Hits@Top K on the benchmark ogbl-ddi dataset. For calculating the Hits@Top K, we use K=20 as recommended by the OGB benchmark.

Model	Hits@Top K(%)
Traditional Configuration Model	**0.99 ± 0.00**
Duplex Configuration Model	**0.99 ± 0.00**
AGDN	0.95 ± 0.01
PSG	0.93 ± 0.01
PLNLP	0.91 ± 0.03

3.1 Topological Sense Features

We make an effort to interpret the shortcuts learning observed in the previous section by identifying the contribution of the first-order network property node degree, and higher-order network properties like clustering coefficient, the number of triangles containing a node, nearest neighbor degree, and betweenness centrality in transductive link prediction. For the ogbl-ddi graph, we use the train graph to compute the aforementioned features for each node. Then, we run a logistic regression using these node features. We apply log transformation and normalization on each feature to ensure uniform scaling. In Figure 3, we visualize the contribution of these topological features in link prediction. The degree of the nearest neighbors of a node corresponds to the largest coefficient in the logistic regression. The coefficient quantifies the importance of the feature in the link prediction task. When the nearest neighbor degree is removed, degree of a node drives the link prediction task, which we observe in the excellent performances of the configuration models. Furthermore, we perform an ablation study to investigate the test performance contributed by each of these topological features (Table 2).

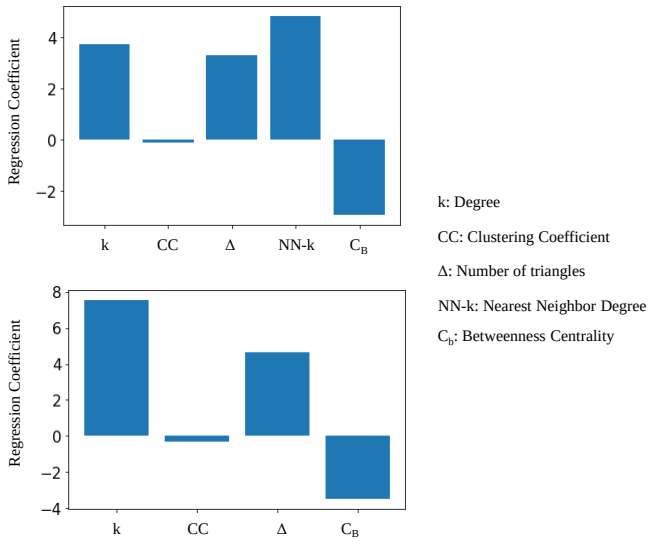

Figure 3: Contribution of the first order and higher order network features in link prediction.

3.2 Degree Bias in Topological Shortcuts

Topological shortcut is associated with the degree distribution of the training graph. Power-law degree distributions [7] create high-degree nodes a.k.a. hubs in the graphs, which contribute to the majority of the edges in train, validation, and test datasets. Link prediction models learn mainly from the degree information of the hubs and make accurate predictions for the links associated with the hubs, producing excellent test performance. Figure 4 shows the test performance of a multi-layer perceptron (MLP, Table 5) on the ogbl-ddi dataset across different degree bins. The low-degree

Table 2: Node degree and the triangle count for each node contribute the most to the link prediction test performance in ogbl-ddi. Here k represents the degree of a node, CC is the local clustering coefficient, Δ is the number of triangle containing a given node, and C_B is betweenness centrality. We use the benchmark train-validation-test split from OGB.

Features	AUROC	AUPRC
k+CC+Δ+C_B	1.0	0.99
CC+Δ+C_B	0.98	0.98
CC+C_B	0.57	0.62

nodes, having degrees less than 100, are encountered less by the model while training, and hence we observe poor test performance for these nodes. The model learns less about the nodes with less topological information, and in the semi-inductive and inductive settings, the model leveraging topological shortcuts fails to make link predictions on the newly arrived nodes with no topological information. Hence, learning the latent graph generation mechanism becomes important in these scenarios and is the key motivation of this work.

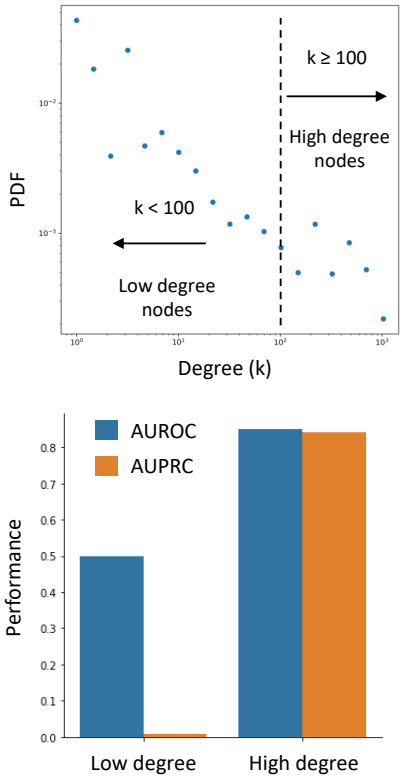

Figure 4: Transductive test performance across different degree bins for ogbl-ddi. The low-degree nodes contribute less to topological shortcuts, producing worse test performance compared to the high-degree nodes a.k.a the hubs.

4 STATE-OF-THE-ART MODELS IN INDUCTIVE LINK PREDICTION

After diving deep into the limitations of transductive tests and acknowledging the importance of semi-inductive and inductive link prediction, we now explore the later settings in detail. We first address the limitations of the existing benchmarks by designing methods for inductive tests on the OGB link prediction benchmark datasets. We use a random node split [72] to define inductive tests on the OGB benchmark datasets. Next, we show that the state-of-the-art link prediction models perform poorly in inductive tests, and achieve lower performances than a simple MLP trained on the node attributes. Non-overlapping topological information between the train and the test graphs in the inductive setting compels the models to only use the node attributes for making meaningful link predictions on the nodes unseen during training.

4.1 Inductive Tests on Static Graphs

Traditionally the train-validation-test split in link prediction tasks is obtained via random edge split [19, 26]. Creating an inductive test scenario by combining the random edge split with the removal of the overlapping nodes between the train and the test graphs eliminates the majority of the links from the training dataset (see here). Inspired by GraIL [72], we use a random node split, which creates inductive test scenarios for any graph dataset. The algorithm for random node split is as follows:

(1) Randomly split the nodes of the original graph V into three groups V_{train}, $V_{validation}$, and V_{test}. We divide the nodes at a 80:10:10 ratio for the train, the validation, and the test datasets, respectively.
(2) Obtain the subgraphs G_{train}, $G_{validation}$, and G_{test} induced by the node sets V_{train}, $V_{validation}$, and V_{test}, respectively.

We observe that the number of edges lost in the random node split is significantly less compared to the random edge split method. We use random node split to design inductive tests on the OGB link prediction datasets. We summarize the number of nodes and edges for each of the undirected link prediction datasets in Table 3 and Table 4, respectively.

Table 3: Number of nodes in different OGB datasets for inductive tests.

Dataset	Train Nodes	Validation Nodes	Test Nodes
ogbl-ppa	461,031	57,629	57,629
ogbl-collab	188,694	23,587	23,587
ogbl-ddi	3,413	427	427

4.2 State-of-the-art Link Prediction Models Fail in Inductive Tests

In order to show the importance of the node attributes in inductive link prediction, we use OGB-defined MLPs (Table 5) which takes the concatenated attributes of the two nodes at the end of each edge as input and then compare its performance with PLNLP [80], a state-of-the-art model on the OGB link prediction leaderboard. For ogbl-ppa, we use the OGB-defined 58-dimensional one-hot feature vectors as the node attributes. These indicate the species associated with the proteins. For ogbl-collab, we use 128-dimensional features obtained by averaging the word embeddings of papers published by the authors. For ogbl-ddi, we use 300-dimensional Mol2vec [35] embeddings of the drug structures. Table 6 summarizes our observations. Inductive link prediction performances are significantly lower than transductive performances for PLNLP, and the MLP performs better in the inductive test compared to this state-of-the-art model.

5 THEORETICAL BACKGROUND

In this section, we develop the theoretical background for the proposed node attribute engineering in order to improve semi-inductive and inductive link prediction performance. Following the standard framework from statistical learning theory [66, 81], we formalize the semi-inductive and inductive link prediction performance as the generalization power of a link prediction model. We denote the instance space characterizing the edges with E and the link prediction hypothesis space with W. The non-negative loss function is $l : W \times E \longrightarrow \mathbf{R}^+$. A link prediction algorithm takes as input n edges as the n-tuple $E = (e_1, ..., e_n)$, which follow an unknown distribution with mean μ, characterizing the latent generation process of the graph. We can write the population risk (quantifies the loss) of the link prediction hypothesis $w \in W$ as:

$$L_\mu(w) \triangleq \mathbf{E}[l(w, E)] = \int_E l(w, e)\mu(de) \tag{2}$$

The μ in the above formulation is tied to the graph generation process. In the inductive link prediction scenario, the generalization error is measured by the ability of the link prediction algorithm to learn the graph generation mechanism from the finite set of observed edges E. The generalization error on μ is measured by $L_\mu(w) - L_E(w)$. The expected value of this generalization error is of the form:

$$gen(\mu, P_{w|E}) \triangleq \mathbf{E}[L_\mu(w) - L_E(w)], \tag{3}$$

i.e., it measures the gap between learning the latent generation mechanism and learning from the finite set of edges.

$$\mathbf{E}[L_\mu(w)] = \mathbf{E}[L_E(w)) + gen(\mu, P_{w|E}] \tag{4}$$

Theorem: If $L(w)$ is σ-subgaussian under $P_{E,w} = \mu^{\bigotimes n} \bigotimes P_{w|E}$, where $\bigotimes$ represents the product of two marginal distributions, then

$$|\mathbf{E}[L_\mu(w)] - \mathbf{E}[L_E(w)]| \leq \sqrt{2\sigma^2 I(E; w)}$$

where $I(E; w)$ quantifies the information shared between the edges and the information learned by the link prediction model.

Proof. Using Corollary 4.15 in [10] for the duality of the entropy of general random variables, we can write the following measure-theoretic bounds:

$$I(E; w) = \sup\{\int_\mu L(w)d\mu - \log \int_E \exp(L(w))de\} \tag{5}$$

Now, replacing the integrals with expectations and replacing the supremum with $\geq$, we get:

Table 4: Edges in OGB link prediction datasets for inductive tests.

Dataset	Train Edges	Validation Edges	Test Edges	Edges lost
ogbl-ppa	19,460,915	296,195	303,563	10,265,600
ogbl-collab	821,974	12,739	12,826	437,926
ogbl-ddi	864,478	13,869	11,433	445,109

Table 5: Description of the MLPs used in the experiments.

Dataset	Layers	Parameters	Hidden Channels	Dropout	Batch size	Learning rate	Epochs
ogbl-ppa	3	113,921	256	0.1	65,536	0.01	20
ogbl-collab	3	99,073	256	0.1	65,536	0.01	200
ogbl-ddi	3	99,073	256	0.1	65,536	0.01	100

Table 6: Inductive test performance of PLNLP is significantly lower than its transductive test performance and is worse than an MLP trained on only the node attributes. For Hits@Top K, we use the default K=100, 50, and 20 for ogbl-ppa, ogbl-collab, and ogbl-ddi, respectively. We run a 5-fold cross-validation in each case.

Dataset	PLNLP in Transductive Test Hits@Top K(%)	PLNLP in Inductive Test Hits@Top K(%)	MLP on node attributes Hits@Top K(%)
ogbl-ppa	32.38 ± 2.58	0.09 ± 0.03	0.39 ± 0.03
ogbl-collab	70.59 ± 0.29	11.56 ± 0.93	36.44 ± 3.11
ogbl-ddi	90.88 ± 3.13	0.01 ± 0.02	0.39 ± 0.02

$$I(E;w) \geq \mathbf{E}[\lambda L_\mu(w)] - \log \mathbf{E}[e^{\lambda L_E(w)}] \geq \lambda(\mathbf{E}[L_\mu(w))] - \mathbf{E}[L_E(w)]) - \frac{\lambda^2\sigma^2}{2} \tag{6}$$

where $\lambda \in \mathbf{R}$ and the second step is derived from the subgaussian nature of $L(w)$ (see section 7.3 for the empirical validation of the subgaussain nature of the loss function in link prediction):

$$\log \mathbf{E}[e^{\lambda(L_E(w) - \mathbf{E}[L_E(w)]}] \leq \frac{\lambda^2\sigma^2}{2}, \forall \lambda \in \mathbf{R} \tag{7}$$

The inequality in Equation 6 generates a non-negative parabola in λ, which has a non-positive discriminant. Thus, we get:

$$|\mathbf{E}[L_\mu(w)] - \mathbf{E}[L_E(w)]| \leq \sqrt{2\sigma^2 I(E;w)} \tag{8}$$

□

Rewriting the above inequality in terms of the generalization error of the link prediction models, we get:

$$|gen(\mu, P_{w|E})| \leq \sqrt{2\sigma^2 I(E;w)} \tag{9}$$

$I(E;w)$ measures the mutual information shared between the link prediction hypothesis and the training data. The link prediction hypothesis w has two components: w_t, which leverages the graph topology, and w_a associated with the node attributes. Thus, we can distribute the mutual information as follows:

$$I(E;w) = I(E;w_t,w_a) = I(w_t,w_a) - I(w_t,w_a|E) \tag{10}$$

In order to minimize the generalization error of the link prediction model from Eq. 9, we need to minimize $I(w_t,w_a)$, i.e., create node attributes that share the least mutual information with the graph topology, independent of the training dataset E. In other words, in order to minimize the generalization loss, and improve semi-inductive and inductive link prediction performance, we need to learn beyond the graph topology, i.e., leverage the node attributes. Combining unlabeled data in the form of unsupervised pre-training has proven results in improving the generalization performance when limited labeled data is available for training [8]. In a similar fashion, by pre-training the node attributes on a large corpus, we improve the generalizability of the link prediction models on the unseen nodes. Intuitively, pre-training of the node attributes enables the link prediction models to reach a minima of the loss function which is suitable for quicker convergence and better generalization [16].

Now, we develop a method to measure the information contained in different types of node attributes using unsupervised clustering and Davies-Bouldin score [14] to identify the attributes most suitable for inductive link prediction. Furthermore, we use adjusted mutual information to quantify the shared information between graph topology and node attributes to validate the theoretical background developed in Section 5.

6 EXPERIMENTAL METHODOLOGY

Many real-world graphs are naturally organized into community structures. In social networks, communities are often based on different interest groups. Protein-protein interaction networks organize the protein structures into communities based on their functions in metabolism [62]. Community detection is a well-explored problem in network science [56]. Various community detection algorithms cluster nodes into meaningful communities based on the graph topology and the node attributes. Locally dense subgraphs

often form communities, which traditional community detection algorithms can detect [24, 63]. Unsupervised clustering on the node attributes helps in detecting communities beyond the topology of the graphs [83]. We use Davies-Bouldin score [14], a cluster separation measure, to quantify the quality of node clusters obtained from unsupervised clustering using only node features. This approach measures the information contained in these node features. Furthermore, with UPNA, when we pre-train the node attributes in an unsupervised fashion on a corpus different from the training graph, the node attributes produce the most separable unsupervised clusters (quantified by Davies-Bouldin score), contain more orthogonal information compared to the graph topology (quantified in terms of adjusted mutual information), and improve generalizability in link prediction (measured in terms of inductive test performance).

We study the quality of the unsupervised node clusters under these scenarios:

- Using Node2vec as the nodes attributes, which encodes the topology of the graph.
- Pre-train the node attributes independently of the training graph.
- Randomly shuffle the pre-trained node attributes for each node individually.
- Replace the pre-trained node attributes with random entries drawn from a uniform distribution, removing any information contained in them.

The random and shuffled versions of the node attributes help us unveil with granularity, the relation between the information contained in the attributes and the inductive link prediction performance. We run the k-means [47] algorithm on the node attributes to obtain unsupervised node clusters. Then we compute the Davies-Bouldin score for each of the node attributes listed above. A lower Davies-Bouldin score implies that the average similarity between clusters is lower and the clustering quality is better. Furthermore, we measure adjusted mutual information (AMI) [74] between the clusters obtained using the pre-trained node attributes and the ones obtained using Node2Vec to quantify $I(w_t, w_a)$ from Section 5. Low AMI values indicate that the node attributes contain information disjoint with the graph topology, and are thus suitable for inductive prediction.

7 RESULTS

We now explore the developed methodology on multiple link prediction benchmark datasets and compare *UPNA* with state-of-the-art models in inductive link prediction.

7.1 Static Graphs

We train the node attributes of the graphs in OGB in an unsupervised manner. For ogbl-ppa, we use 100-dimensional ProtVec [4] vectors as the pre-trained node attributes. The corpus used in the training of the ProtVec model includes 546,790 amino acid sequences from the Swiss-Prot database [5]. For ogbl-collab, we use 128-dimensional Word2vec embeddings on the papers for each author. The Word2Vec training corpus uses Google news articles with approximately 6B tokens [55]. For ogbl-ddi, we use the pre-trained 300-dimensional Mol2vec [35] embeddings as the node attributes. The Mol2vec training corpus includes 19.9M chemicals from ZINC [34] and ChEMBL [22] libraries.

We consider the effect of shuffling the components of each node attribute vector for each node individually to disrupt the information contained in them. This process keeps the distribution of the attributes unchanged for each node. In another experiment, we replace the node attributes with uniform random entries. Comparing the Davies-Bouldin scores for all these scenarios shows us that the pre-trained node attributes produce the most meaningful node clustering with the lowest (lower is better) Davies-Bouldin score (see Figure 5). Hence, the pre-trained node attributes are the most informative node features beyond the graph topology, and the most suited for inductive tests. Moreover, in Table 7 we observe low AMI between the pre-trained node attributes and Node2vec, signifying low information shared between the node attributes and the graph topology, and hence low $I(w_t, w_a)$. This confirms that the pre-trained node attributes accommodate minimal topological information from the train graph and improve the generalization power of the link prediction models according to Eq. 9.

Table 7: We compare the adjusted mutual information between the unsupervised clusters obtained using only Node2vec and the pre-trained node attributes. Node2vec uses the graph topology to create the node clusters. Low AMI values ($\ll 1$) indicate that the pre-trained attributes share minimal information with the graph topology.

Dataset	$I(w_t, w_a)$
ogbl-ppa	0.17
ogbl-collab	0.08
ogbl-ddi	0.22

We use the node features described above for inductive link prediction on the OGB benchmark. We use the MLP architectures described in Table 5. In Figure 6, we observe that the pre-trained node attributes, for which the Davies-Bouldin scores are the lowest, (i.e., the node attributes which create the best-unsupervised node clustering) perform the best in inductive link prediction.

7.2 Temporal Graphs

We test our proposed method UNAP on temporal networks in predicting links for newly arrived nodes as formulated in subsection 2.2. We use the open-source reddit hyperlink network dataset [39]. The dataset contains reddit data for 3 years, starting from 2014 to 2017. The nodes here are the subreddit communities and the links are the posts shared between two communities. The pre-trained node attributes are generated from the content of the posts using GloVe word embeddings trained on 840B words [59]. We use a 3-layer MLP as the decoder with hidden layer sizes of 100, a learning rate of 0.001, epochs of 200, ReLU activation functions, and ADAM solver. We observe that the pre-trained node attributes with the lowest Davies-Bouldin score perform the best in predicting the links on the newly arrived nodes across different years (Table 8). Furthermore, we compare *UPNA* with DyHATR [82], a state-of-the-art temporal link prediction model in inductive link prediction. The observations are summarized in Table 8.

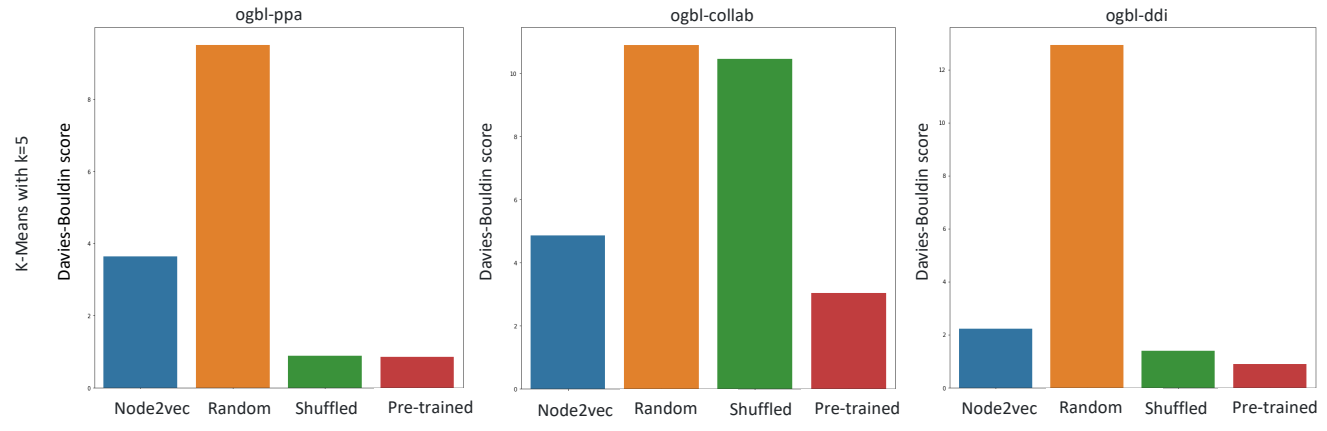

Figure 5: Comparing the quality of the unsupervised clusters using Davies-Bouldin score for various node features on the OGB link prediction benchmarks. The pre-trained node attributes show the lowest Davies-Bouldin score in all of the graph datasets, and hence are the most suitable for inductive link prediction tasks. Here, we run the K-means algorithm with K=5. Similar observations are made for K=10.

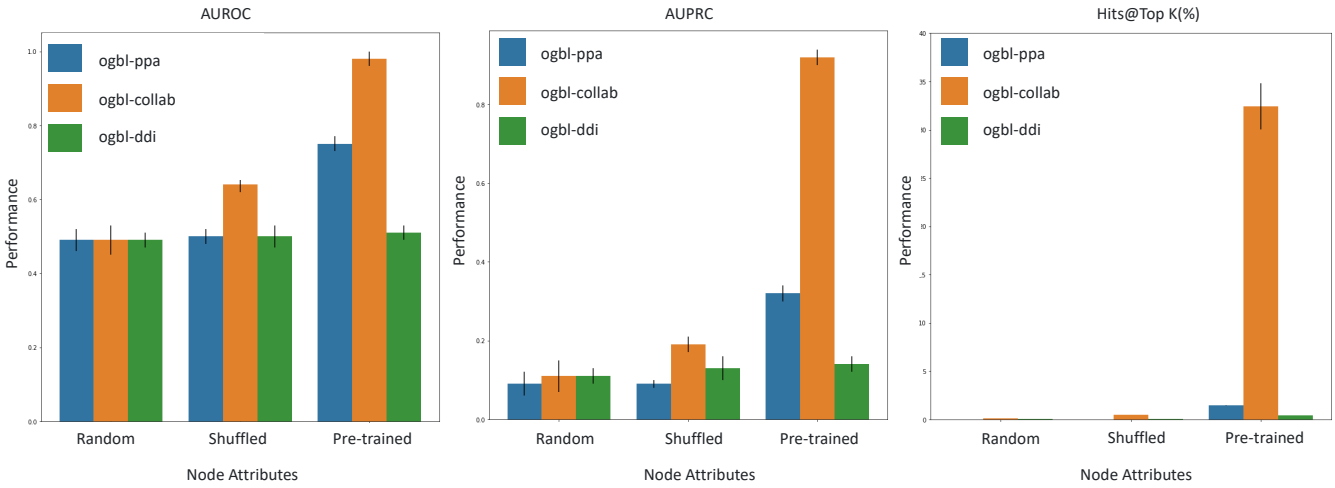

Figure 6: We compare AUROC, AUPRC and Hits@Top K performances for different node attributes in inductive tests. The pre-trained node attributes show the highest performance in all of the OGB link prediction datasets.

7.3 Subgaussian nature of the loss function in link prediction

We empirically explore the subgaussian nature of the loss function of the link prediction model used on the subreddit network. Subgaussianity of the loss function yields Eq. 7, which is used to derive the parabolic form in Eq. 6. We show in Figure 7 that the loss function during training of the link prediction model decays faster than the Gaussian tail characterized by $L = Ae^{-\sigma x^2}$ where $A = 0.78$ and $\sigma = 0.001$ (obtained from an exponential fit). Thus, the loss function is σ-subgaussian in nature, with $\sigma = 0.001$.

8 RELATED WORK

Machine learning models use both the graph topology and the node attributes in link prediction [3]. When the train and the test graphs share similar topologies, the topological information is of more importance compared to the node attributes. Existing inductive link prediction methods like GraphSAGE [26] perform well in inductive tests when the links associated with the newly arrived nodes are known, and the test graph shares similar topological features as the graph used in training. As a random edge split creates the train and the test graphs with similar topologies, using simple topological features on the nodes and the links (like degree, centrality etc.) are sufficient to predict the unobserved links [23].

Table 8: We report the link prediction performance on the newly arrived nodes for different temporal instances of the subreddit network. We observe that the link prediction performance is related to the Davies-Bouldin scores of the node attributes (for K-means clustering with $K = 5$). The lower the score, the more information contained in the attributes, and the higher the performance. The pre-trained node attributes provide the best link prediction performance.

Node Attributes	2014-2015			2016-2017			2017-2018		
	DB Score	AUROC	AUPRC	DB Score	AUROC	AUPRC	DB Score	AUROC	AUPRC
Pre-trained + MLP	**2.9**	**0.70**	**0.66**	**0.2**	**0.63**	**0.60**	**2.8**	**0.69**	**0.65**
Shuffled + MLP	20.5	0.66	0.62	15.04	0.59	0.55	16.9	0.60	0.56
Random + MLP	21.9	0.5	0.5	22.3	0.5	0.5	21.3	0.51	0.51
DyHATR	-	0.45	0.25	-	0.45	0.48	-	0.46	0.28

Joachims [37] first mentioned the difficulty of inductive tests compared to transductive tests in a text classification setting. This understanding is valid for other domains of machine learning, and can be extended to graph machine learning, particularly link prediction. Planetoid [84], GraIL [72], and GraphSAGE [26] are among the most recognized methods for inductive link prediction which require link information for the newly arrived nodes and leverage topological shortcuts. GraphSAGE learns the neighborhood information for each node in training to make predictions using the neighborhoods of the newly arrived nodes in inductive tests. DEAL [28] uses both topological information and node attributes in making link predictions in both transductive and inductive settings. Structure Enhanced Graph neural network (SEG) uses a simple one-layer GCN to encode the topology of the training graph, which combined with a simple multilayer perceptron (MLP) on the node attributes significantly improves the transductive test performance [3]. A recently proposed open challenge for inductive link prediction on Knowledge Graphs [19] has inspired multiple state-of-the-art inductive link prediction models like CascadER [64] and Inductive NodePiece [20].

The importance of node attributes in graph stream problems, dynamic network research [36, 41], and cold-start setting [21, 42] has been studied extensively. Pre-training has recently been used in dynamic network research [67] and cold-start problem [27, 44, 79], but in training the parameters of the GNN models and not the node attributes. Our method, *UPNA*, introduces pre-training for node attributes and learns representations over large corpora beyond the training graph. Due to the generalizable embeddings, *UPNA* can be integrated into the existing models for link prediction in dynamic networks and cold-start problems to improve generalizability.

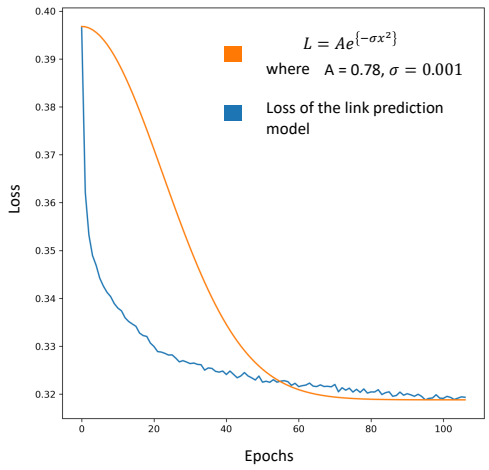

Figure 7: The loss function in link prediction decays faster than a Gaussian tail, creating a subgaussian behavior.

9 CONCLUSION AND FUTURE WORK

We showed how link prediction models leverage the topology of the train graph to achieve excellent transductive test performance. We then established the importance of inductive link prediction and formulated the prediction task on both static and temporal graphs. We observed that the performance of the state-of-the-art link prediction models reduces significantly in inductive tests compared to the transductive setting. These performances are comparable, and sometimes lower than that of a simple MLP on the node attributes. Next, we theoretically justified how inductive link prediction is associated with the information shared between the node attributes and the graph topology. Furthermore, we have developed a method to quantify the goodness of the node attributes, and have experimentally shown that the pre-trained node attributes are the most suitable for improving the generalizability in link prediction. In future work, we plan to incorporate the pre-trained node attributes in multiple state-of-the-art link prediction models for improved overall link prediction performance. We also plan to incorporate the Davies-Bouldin score on the node attributes in the pre-training objective function for obtaining the optimal set of node attributes. Finally, exploring the effect of the size of the training corpus of the pre-trained node attributes on inductive link prediction performance would bring more insight into the hypotheses proposed by Erhan et al. on unsupervised pre-training [16]. Overall, our work develops a method for selecting the best node attributes for improving the generalizability of the link prediction models for newly arrived nodes.

Our software and information about the data used in the experiments are at https://anonymous.4open.science/r/inductive-link-prediction-6B65.

REFERENCES

[1] Lada A Adamic and Eytan Adar. 2003. Friends and neighbors on the Web. *Social Networks* 25, 3 (July 2003), 211–230. https://doi.org/10.1016/s0378-8733(03)00009-1

[2] Florian Adriaens, Alexandru Mara, Jefrey Lijffijt, and Tijl De Bie. 2020. Block-Approximated Exponential Random Graphs. In *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE. https://doi.org/10.1109/dsaa49011.2020.00019

[3] Baole Ai, Zhou Qin, Wenting Shen, and Yong Li. 2022. Structure Enhanced Graph Neural Networks for Link Prediction. https://doi.org/10.48550/ARXIV.2201.05293

[4] Ehsaneddin Asgari and Mohammad R. K. Mofrad. 2015. Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics. *PLOS ONE* 10, 11 (Nov. 2015), e0141287. https://doi.org/10.1371/journal.pone.0141287

[5] A Bairoch. 1996. The SWISS-PROT protein sequence data bank and its new supplement TREMBL. *Nucleic Acids Research* 24, 1 (Jan. 1996), 21–25. https://doi.org/10.1093/nar/24.1.21

[6] Brian Ball and M.E.J. Newman. 2013. Friendship networks and social status. *Network Science* 1, 1 (apr 2013), 16–30. https://doi.org/10.1017/nws.2012.4

[7] Albert-László Barabási and Réka Alber. 1999. Emergence of Scaling in Random Networks. *Science* 286, 5439 (Oct. 1999), 509–512. https://doi.org/10.1126/science.286.5439.509

[8] Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory - COLT' 98*. ACM Press, Madison Wisconsin, USA, 92–100. https://doi.org/10.1145/279943.279962

[9] Stephen Bonner, Ufuk Kirik, Ola Engkvist, Jian Tang, and Ian P Barrett. 2022. Implications of topological imbalance for representation learning on biomedical knowledge graphs. *Briefings in Bioinformatics* 23, 5 (07 2022), 1–14. https://doi.org/10.1093/bib/bbac279 arXiv:https://academic.oup.com/bib/article-pdf/23/5/bbac279/45937607/sup_main_bbac279.pdf bbac279.

[10] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. 2013. 83Basic Information Inequalities. In *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, Oxford, England. https://doi.org/10.1093/acprof:oso/9780199535255.003.0004 arXiv:https://academic.oup.com/book/0/chapter/195073150/chapter-pdf/43899191/acprof-9780199535255-chapter-04.pdf

[11] Shaosheng Cao, Wei Lu, and Qiongkai Xu. 2015. GraRep: Learning Graph Representations with Global Structural Information. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management* (Melbourne, Australia) *(CIKM '15)*. Association for Computing Machinery, New York, NY, USA, 891–900. https://doi.org/10.1145/2806416.2806512

[12] Ayan Chatterjee, Robin Walters, Zohair Shafi, Omair Shafi Ahmed, Michael Sebek, Deisy Gysi, Rose Yu, Tina Eliassi-Rad, Albert-László Barabási, and Giulia Menichetti. 2023. Improving the generalizability of protein-ligand binding predictions with AI-Bind. *Nature Communications* 14, 1 (April 2023). https://doi.org/10.1038/s41467-023-37572-z

[13] Xu Chen, Siheng Chen, Huangjie Zheng, Jiangchao Yao, Kenan Cui, Ya Zhang, and Ivor W. Tsang. 2019. Node Attribute Generation on Graphs. https://doi.org/10.48550/ARXIV.1907.09708

[14] David L. Davies and Donald W. Bouldin. 1979. A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1, 2 (April 1979), 224–227. https://doi.org/10.1109/tpami.1979.4766909

[15] Vijay Prakash Dwivedi, Chaitanya K. Joshi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. 2020. Benchmarking Graph Neural Networks. https://doi.org/10.48550/ARXIV.2003.00982

[16] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. 2010. Why Does Unsupervised Pre-training Help Deep Learning? *Journal of Machine Learning Research* 11, 19 (2010), 625–660. http://jmlr.org/papers/v11/erhan10a.html

[17] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. 2019. Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems* (Copenhagen, Denmark) *(RecSys '19)*. Association for Computing Machinery, New York, NY, USA, 101–109. https://doi.org/10.1145/3298689.3347058

[18] Agata Fronczak. 2012. Exponential random graph models. https://doi.org/10.48550/ARXIV.1210.7828

[19] Mikhail Galkin, Max Berrendorf, and Charles Tapley Hoyt. 2022. An Open Challenge for Inductive Link Prediction on Knowledge Graphs. https://doi.org/10.48550/ARXIV.2203.01520

[20] Mikhail Galkin, Etienne Denis, Jiapeng Wu, and William L. Hamilton. 2021. NodePiece: Compositional and Parameter-Efficient Representations of Large Knowledge Graphs. https://doi.org/10.48550/ARXIV.2106.12144

[21] Zeno Gantner, Lucas Drumond, Christoph Freudenthaler, Steffen Rendle, and Lars Schmidt-Thieme. 2010. Learning Attribute-to-Feature Mappings for Cold-Start Recommendations. In *2010 IEEE International Conference on Data Mining*. 176–185. https://doi.org/10.1109/ICDM.2010.129

[22] A. Gaulton, L. J. Bellis, A. P. Bento, J. Chambers, M. Davies, A. Hersey, Y. Light, S. McGlinchey, D. Michalovich, B. Al-Lazikani, and J. P. Overington. 2011. ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Research* 40, D1 (Sept. 2011), D1100–D1107. https://doi.org/10.1093/nar/gkr777

[23] Amir Ghasemian, Homa Hosseinmardi, Aram Galstyan, Edoardo M. Airoldi, and Aaron Clauset. 2020. Stacking models for nearly optimal link prediction in complex networks. *Proceedings of the National Academy of Sciences* 117 (2020), 23393–23400. Issue 38. https://www.pnas.org/doi/pdf/10.1073/pnas.1914950117

[24] M. Girvan and M. E. J. Newman. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* 99, 12 (June 2002), 7821–7826. https://doi.org/10.1073/pnas.122653799

[25] Aditya Grover and Jure Leskovec. 2016. Node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Francisco, California, USA) *(KDD '16)*. Association for Computing Machinery, New York, NY, USA, 855–864. https://doi.org/10.1145/2939672.2939754

[26] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. https://doi.org/10.48550/ARXIV.1706.02216

[27] Bowen Hao, Hongzhi Yin, Jing Zhang, Cuiping Li, and Hong Chen. 2023. A Multi-Strategy-Based Pre-Training Method for Cold-Start Recommendation. *ACM Trans. Inf. Syst.* 41, 2, Article 31 (jan 2023), 24 pages. https://doi.org/10.1145/3544107

[28] Yu Hao, Xin Cao, Yixiang Fang, Xike Xie, and Sibo Wang. 2021. Inductive Link Prediction for Nodes Having Only Attribute Information. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI'20)*. ACM, Yokohama, Yokohama, Japan, Article 168, 7 pages.

[29] Petter Holme and Jari Saramäki. 2012. Temporal networks. *Physics Reports* 519, 3 (oct 2012), 97–125. https://doi.org/10.1016/j.physrep.2012.03.001

[30] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. 2020. Open Graph Benchmark: Datasets for Machine Learning on Graphs. https://doi.org/10.48550/ARXIV.2005.00687

[31] Kexin Huang, Tianfan Fu, Lucas M Glass, Marinka Zitnik, Cao Xiao, and Jimeng Sun. 2020. DeepPurpose: a deep learning library for drug–target interaction prediction. *Bioinformatics* 36, 22-23 (Dec. 2020), 5545–5547. https://doi.org/10.1093/bioinformatics/btaa1005

[32] Kexin Huang, Cao Xiao, Lucas M Glass, and Jimeng Sun. 2020. MolTrans: Molecular Interaction Transformer for drug–target interaction prediction. *Bioinformatics* 37, 6 (Oct. 2020), 830–836. https://doi.org/10.1093/bioinformatics/btaa880

[33] Qian Huang, Horace He, Abhay Singh, Ser-Nam Lim, and Austin R. Benson. 2020. Combining Label Propagation and Simple Models Out-performs Graph Neural Networks. https://doi.org/10.48550/ARXIV.2010.13993

[34] John J. Irwin, Teague Sterling, Michael M. Mysinger, Erin S. Bolstad, and Ryan G. Coleman. 2012. ZINC: A Free Tool to Discover Chemistry for Biology. *Journal of Chemical Information and Modeling* 52, 7 (June 2012), 1757–1768. https://doi.org/10.1021/ci3001277

[35] Sabrina Jaeger, Simone Fulle, and Samo Turk. 2018. Mol2vec: Unsupervised Machine Learning Approach with Chemical Intuition. *Journal of Chemical Information and Modeling* 58, 1 (Jan. 2018), 27–35. https://doi.org/10.1021/acs.jcim.7b00616

[36] Maosheng Jiang, Yonxiang Chen, and Ling Chen. 2015. Link Prediction in Networks with Nodes Attributes by Similarity Propagation. arXiv:1502.04380 [cs.SI]

[37] Thorsten Joachims. 1999. Transductive Inference for Text Classification Using Support Vector Machines. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML '99)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 200–209.

[38] Ajay Kumar, Shashank Sheshar Singh, Kuldeep Singh, and Bhaskar Biswas. 2020. Link prediction techniques, applications, and performance: A survey. *Physica A: Statistical Mechanics and its Applications* 553 (Sept. 2020), 124289. https://doi.org/10.1016/j.physa.2020.124289

[39] Srijan Kumar, William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2018. Community Interaction and Conflict on the Web. In *Proceedings of the 2018 World Wide Web Conference* (Lyon, France) *(WWW '18)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 933–943. https://doi.org/10.1145/3178876.3186141

[40] T. Jaya Lakshmi and S. Durga Bhavani. 2021. Link Prediction Approach to Recommender Systems. https://doi.org/10.48550/ARXIV.2102.09185

[41] Jundong Li, Kewei Cheng, Liang Wu, and Huan Liu. 2018. Streaming Link Prediction on Dynamic Attributed Networks. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining* (Marina Del Rey, CA, USA) *(WSDM '18)*. Association for Computing Machinery, New York, NY, USA, 369–377. https://doi.org/10.1145/3159652.3159674

[42] Shijun Li, Wenqiang Lei, Qingyun Wu, Xiangnan He, Peng Jiang, and Tat-Seng Chua. 2021. Seamlessly Unifying Attributes and Items: Conversational Recommendation for Cold-Start Users. *ACM Trans. Inf. Syst.* 39, 4, Article 40 (aug 2021), 29 pages. https://doi.org/10.1145/3446427

[43] David Liben-Nowell and Jon Kleinberg. 2007. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology* 58, 7 (2007), 1019–1031. https://doi.org/10.1002/asi.20591

[44] Zemin Liu, Xingtong Yu, Yuan Fang, and Xinming Zhang. 2023. GraphPrompt: Unifying Pre-Training and Downstream Tasks for Graph Neural Networks. arXiv:2302.08043 [cs.LG]

[45] Oriol Lordan and Jose M. Sallan. 2020. Dynamic measures for transportation networks. *PLOS ONE* 15, 12 (Dec. 2020), e0242875. https://doi.org/10.1371/journal.pone.0242875

[46] Jingsong Lv, Zhao Li, Hongyang Chen, Yao Qi, and Chunqi Wu. 2022. Path-aware Siamese Graph Neural Network for Link Prediction. https://doi.org/10.48550/ARXIV.2208.05781

[47] J. B. MacQueen. 1967. Some Methods for Classification and Analysis of MultiVariate Observations. *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability* 1 (1967), 281–297.

[48] Ivan Maksimov, Rodrigo Rivera-Castro, and Evgeny Burnaev. 2020. Addressing Cold Start in Recommender Systems with Hierarchical Graph Neural Networks. https://doi.org/10.48550/ARXIV.2009.03455

[49] Alexandru Cristian Mara, Jefrey Lijffijt, and Tijl De Bie. 2022. An Empirical Evaluation of Network Representation Learning Methods. *Big Data* (March 2022). https://doi.org/10.1089/big.2021.0107

[50] Alexandru Cristian Mara, Jefrey Lijffijt, and Tijl de Bie. 2020. Benchmarking Network Embedding Models for Link Prediction: Are We Making Progress?. In *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE. https://doi.org/10.1109/dsaa49011.2020.00026

[51] Giulia Menichetti. 2022. An AI pipeline to investigate the binding properties of poorly annotated molecules. *Nature Reviews Physics* 4, 6 (May 2022), 359–359. https://doi.org/10.1038/s42254-022-00471-1

[52] Giulia Menichetti, Ginestra Bianconi, Gastone Castellani, Enrico Giampieri, and Daniel Remondini. 2015. Multiscale characterization of ageing and cancer progression by a novel network entropy measure. *Molecular bioSystems* 11, 7 (2015), 1824–31. http://pubs.rsc.org/en/content/articlehtml/2015/mb/c5mb00143a

[53] Giulia Menichetti and Daniel Remondini. 2014. Entropy of a network ensemble: Definitions and applications to genomic data. *Theoretical Biology Forum* 107, 1-2 (2014), 77–87.

[54] Giulia Menichetti, Daniel Remondini, Pietro Panzarasa, Raúl J. Mondragón, and Ginestra Bianconi. 2014. Weighted Multiplex Networks. *PLoS ONE* 9, 6 (June 2014), e97857. https://doi.org/10.1371/journal.pone.0097857

[55] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. https://doi.org/10.48550/ARXIV.1301.3781

[56] M. E. J. Newman. 2006. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences* 103, 23 (June 2006), 8577–8582. https://doi.org/10.1073/pnas.0601602103

[57] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2016. A Review of Relational Machine Learning for Knowledge Graphs. *Proc. IEEE* 104, 1 (jan 2016), 11–33. https://doi.org/10.1109/jproc.2015.2483592

[58] Giuliano Andrea Pagani and Marco Aiello. 2011. The Power Grid as a Complex Network: a Survey. https://doi.org/10.48550/ARXIV.1105.3338

[59] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, 1532–1543. https://doi.org/10.3115/v1/D14-1162

[60] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: Online Learning of Social Representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA) *(KDD '14)*. Association for Computing Machinery, New York, NY, USA, 701–710. https://doi.org/10.1145/2623330.2623732

[61] Yanjun Qi, Ziv Bar-Joseph, and Judith Klein-Seetharaman. 2006. Evaluation of different biological data and computational classification methods for use in protein interaction prediction. *Proteins: Structure, Function, and Bioinformatics* 63, 3 (Jan. 2006), 490–500. https://doi.org/10.1002/prot.20865

[62] Filippo Radicchi, Claudio Castellano, Federico Cecconi, Vittorio Loreto, and Domenico Parisi. 2004. Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences* 101, 9 (Feb. 2004), 2658–2663. https://doi.org/10.1073/pnas.0400054101

[63] M. Rosvall, D. Axelsson, and C. T. Bergstrom. 2009. The map equation. *The European Physical Journal Special Topics* 178, 1 (Nov. 2009), 13–23. https://doi.org/10.1140/epjst/e2010-01179-1

[64] Tara Safavi, Doug Downey, and Tom Hope. 2022. CascadER: Cross-Modal Cascading for Knowledge Graph Link Prediction. https://doi.org/10.48550/ARXIV.2205.08012

[65] Ingo Scholtes, Nicolas Wider, and Antonios Garas. 2016. Higher-order aggregate networks in the analysis of temporal networks: path structures and centralities. *The European Physical Journal B* 89, 3 (mar 2016), 1–15. https://doi.org/10.1140/epjb/e2016-60663-0

[66] Shai Shalev-Shwartz and Shai Ben-David. 2014. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, New York, USA. I–XVI, 1–397 pages.

[67] Zezhi Shao, Zhao Zhang, Fei Wang, and Yongjun Xu. 2022. Pre-Training Enhanced Spatial-Temporal Graph Neural Network for Multivariate Time Series Forecasting. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Washington DC, USA) *(KDD '22)*. Association for Computing Machinery, New York, NY, USA, 1567–1577. https://doi.org/10.1145/3534678.3539396

[68] Zachary Stanfield, Mustafa Coşkun, and Mehmet Koyutürk. 2017. Drug Response Prediction as a Link Prediction Problem. *Scientific Reports* 7, 1 (Jan. 2017), 1–12. https://doi.org/10.1038/srep40321

[69] Andrew Stolman, Caleb Levy, C. Seshadhri, and Aneesh Sharma. 2022. Classic Graph Structural Features Outperform Factorization-Based Graph Embedding Methods on Community Labeling. In *Proceedings of the 2022 SIAM International on Conference on Data Mining (SDM)*. SIAM, Alexandria, VA, USA, 388–396. https://arxiv.org/abs/2201.08481

[70] Chuxiong Sun, Jie Hu, Hongming Gu, Jinpeng Chen, and Mingchuan Yang. 2020. Adaptive Graph Diffusion Networks. https://doi.org/10.48550/ARXIV.2012.15024

[71] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. LINE: Large-Scale Information Network Embedding. In *Proceedings of the 24th International Conference on World Wide Web* (Florence, Italy) *(WWW '15)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1067–1077. https://doi.org/10.1145/2736277.2741093

[72] Komal K. Teru, Etienne Denis, and William L. Hamilton. 2019. Inductive Relation Prediction by Subgraph Reasoning. https://doi.org/10.48550/ARXIV.1911.06962

[73] Pim van der Hoorn, Gabor Lippner, and Dmitri Krioukov. 2017. Sparse Maximum-Entropy Random Graphs with a Given Power-Law Degree Distribution. *Journal of Statistical Physics* 173, 3-4 (oct 2017), 806–844. https://doi.org/10.1007/s10955-017-1887-7

[74] Nguyen Xuan Vinh, Julien Epps, and James Bailey. 2010. Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. *J. Mach. Learn. Res.* 11 (dec 2010), 2837–2854.

[75] Daixin Wang, Peng Cui, and Wenwu Zhu. 2016. Structural Deep Network Embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. ACM, San Francisco, CA, USA, 1225–1234. https://doi.org/10.1145/2939672.2939753

[76] Kuansan Wang, Zhihong Shen, Chiyuan Huang, Chieh-Han Wu, Yuxiao Dong, and Anshul Kanakia. 2020. Microsoft Academic Graph: When experts are not enough. *Quantitative Science Studies* 1, 1 (Feb. 2020), 396–413. https://doi.org/10.1162/qss_a_00021

[77] Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, Tianjun Xiao, Tong He, George Karypis, Jinyang Li, and Zheng Zhang. 2019. Deep Graph Library: A Graph-Centric, Highly-Performant Package for Graph Neural Networks. https://doi.org/10.48550/ARXIV.1909.01315

[78] Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, Tianjun Xiao, Tong He, George Karypis, Jinyang Li, and Zheng Zhang. 2019. Deep Graph Library: A Graph-Centric, Highly-Performant Package for Graph Neural Networks. https://doi.org/10.48550/ARXIV.1909.01315

[79] Yiqi Wang, Chaozhuo Li, Zheng Liu, Mingzheng Li, Jiliang Tang, Xing Xie, Lei Chen, and Philip S. Yu. 2022. An Adaptive Graph Pre-Training Framework for Localized Collaborative Filtering. *ACM Trans. Inf. Syst.* 41, 2, Article 43 (dec 2022), 27 pages. https://doi.org/10.1145/3555372

[80] Zhitao Wang, Yong Zhou, Litao Hong, Yuanhang Zou, Hanjing Su, and Shouzhi Chen. 2021. Pairwise Learning for Neural Link Prediction. https://doi.org/10.48550/ARXIV.2112.02936

[81] Aolin Xu and Maxim Raginsky. 2017. Information-theoretic analysis of generalization capability of learning algorithms. https://doi.org/10.48550/ARXIV.1705.07809

[82] Hansheng Xue, Luwei Yang, Wen Jiang, Yi Wei, Yi Hu, and Yu Lin. 2020. Modeling Dynamic Heterogeneous Network for Link Prediction Using Hierarchical Attention with Temporal RNN. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part I* (Ghent, Belgium). Springer-Verlag, Berlin, Heidelberg, 282–298. https://doi.org/10.1007/978-3-030-67658-2_17

[83] Jaewon Yang, Julian McAuley, and Jure Leskovec. 2013. Community Detection in Networks with Node Attributes. https://doi.org/10.1109/icdm.2013.167

[84] Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov. 2016. Revisiting Semi-Supervised Learning with Graph Embeddings. https://doi.org/10.48550/ARXIV.1603.08861