Supplementary Material

September 2, 2025

Contents

1 A brief literature review of link prediction methods

In the main paper, we provide a concise literature review of related work on link prediction based on methodological differences. Here, we adopt an alternative taxonomy that categorizes the literature into link prediction in static networks and dynamic networks. Interested readers are referred to more comprehensive surveys [1–7] on this topic under different classification schemes.

1.1 Link prediction in static networks

Link prediction for static networks has a long history. Since Liben-Nowell et al. [8] formulated the link prediction problem and proposed a basic framework based on node similarity, the topic has been extensively studied. We discuss the existing methods from three categories: similarity-based methods, probabilistic and statistical models, dimensionality reduction-based methods, and learning-based approaches.

1.1.1 Similarity-based methods

Similarity-based metrics are the simplest class of methods for link prediction. The similarity between node pairs is typically computed using structural properties of the network. Based on the extent of structural information considered, these methods can be grouped into three categories: local, quasi-local, and global.

Local indices are generally calculated using information about common neighbors and node degree. These indices consider immediate neighbors of a node. Examples of such indices contains common neighbor [9], preferential attachment [10], Adamic/Adar [11], Jaccard Coefficient [12], resource allocation [13], etc. Global similarity indices are computed based on the entire topological structure of a network. While these methods often achieve high prediction accuracy by capturing long-range dependencies between nodes, they typically involve higher computational complexity, making them less practical for large-scale networks. Representative global indices include the Katz Index [14], Random Walk with Restart (RWR) [15], Shortest Path [16], Path Entropy [17], and others. Quasi-local indices achieve a balance between local and global approaches by trading off predictive accuracy and computational cost. These methods generally offer better performance than purely local indices by incorporating a broader range of structural information. However, their performance typically falls short of fully global methods, as they rely on less comprehensive topological information. Representative examples include the Local Path Index [18], Local Random Walk Index [19], and Local Directed Path (LDP) [20].

1.1.2 Probabilistic and statistical models

Probabilistic and statistical models typically assume a known or observable network structure and aim to construct models that fit this structure while estimating parameters using statistical methods. To improve predictive performance, such models often incorporate not only topological information but also node and edge attributes. However, the high computational cost and the requirement for rich attribute data limit the scalability and applicability of probabilistic models in large or dynamic networks.

Wang et al. [21] introduced a local probabilistic model that leverages topological, semantic, and co-occurrence features for link prediction. They proposed the concept of a central neighborhood set and extracted non-derivable frequent itemsets from event logs to serve as input for training a local Markov Random Field (MRF). The model is iteratively optimized using probabilistic inference techniques to estimate the likelihood of potential links. Probabilistic Relational Models (PRMs) [22–24] offer a unified framework for link prediction by incorporating both node and link attributes. They model heterogeneous entities (e.g., persons, articles, institutions) and their relationships by casting link prediction as an attribute inference problem within a relational graph structure. Clauset et al. [25] proposed a hierarchical probabilistic model that learns network structure from data through a tree-based representation. The model estimates link likelihoods based on hierarchical groupings, providing insights into the organization of complex networks. Stochastic Block Models (SBMs) [26, 27] extend the hierarchical structure concept by partitioning nodes into blocks or communities, where the probability of a connection between nodes depends on their respective group memberships. Guimerà et al. [28] utilized SBMs to evaluate the reliability of observed links and to identify potential missing or spurious connections. Exponential Random Graph Models (ERGMs) [29] define a probability distribution over possible network configurations, allowing for the modeling of complex structural dependencies. Several studies [30–32] have applied

ERGMs to discover plausible links by comparing the observed network with random ensembles under predefined structural principles.

1.1.3 Dimensionality reduction-based methods

The curse of dimensionality poses significant challenges in link prediction. To address this, dimensionality reduction techniques such as network embedding and matrix factorization have gained increasing attention.

Network embedding methods learn low-dimensional vector representations of nodes that preserve network proximity. Classical techniques include Laplacian Eigenmaps [33], Locally Linear Embedding (LLE) [34], and Isomap [35]. However, these methods often face scalability issues. Recent advances exploit the sparsity of real-world networks to improve scalability. For example, DeepWalk [36] treats truncated random walks as sentences in a language model to capture higher-order proximities, while Node2vec [37] introduces biased random walks that balance breadth-first and depth-first search strategies, demonstrating improved performance over DeepWalk. Complex embeddings, as introduced by Trouillon et al., leverage matrix and tensor factorization with complex-valued vectors to model diverse binary relations, including symmetric and antisymmetric ones. Menon et al. [38] proposed a supervised matrix factorization framework that addresses class imbalance by optimizing ranking loss and integrating explicit node and link features for improved link prediction. Similar approaches by Chen et al. [39] combined topological and attribute matrices via non-negative matrix factorization.

1.1.4 Learning-based methods

In learning-based methods, link prediction is often formulated as a binary classification problem, where each data point corresponds to a node pair and the label indicates the presence or absence of a link. Various classifiers [40,41] (e.g., SVM, decision trees, naive Bayes) can be employed; however, a key challenge lies in selecting effective features. Most existing works [42] rely on topological features, which are domain-independent and computationally efficient.

In recent years, deep learning techniques have shown great promise in link prediction by learning high-level representations from structural and attribute data. Schlichtkrul et al. developed the Relational Graph Convolutional Network (R-GCN) [43], which models knowledge graphs as directed multigraphs with labeled nodes and edges, enabling link prediction via autoencoding. Kipf et al. proposed the Variational Graph Autoencoder (VGAE) [44], which learns latent node representations from the adjacency and feature matrices of graphs. The input is normalized and self-loops are added to improve stability during training. More recent models, such as the Weisfeiler–Lehman Neural Machine (WLNM) [45], capture higher-order local structures through subgraph extraction and use neural networks to learn link patterns. The SEAL framework [46] further integrates local subgraphs, embeddings, and attributes within a unified GCN-based model, demonstrating superior performance over heuristic and latent feature-based methods.

1.1.5 Summary of static link prediction methods

For Static link prediction, similarity-based methods offer intuitive and computationally efficient solutions by leveraging local to global network structures. Probabilistic models predict links via statistical inference, often integrating topological information along with node or edge attributes to improve accuracy. Dimensionality reduction methods, including network embedding and matrix factorization, effectively address high-dimensionality issues by learning compact node representations. Learning-based approaches, especially those leveraging deep learning, have recently advanced the field by capturing complex patterns and high-order structures, leading to improved prediction accuracy. Together, these diverse methodologies provide a solid foundation for tackling various static link prediction problems.

1.2 Link prediction in dynamic networks

Dynamic link prediction is a fundamental problem in the analysis of evolving networks, with applications in social networks, communication systems, and biological networks. As networks change over time, accurately predicting future links requires capturing both structural and temporal dependencies. To this end, a wide range of learning-based methods have been proposed, which can be broadly categorized into unsupervised and supervised approaches.

1.2.1 Unsupervised learning methods

Unsupervised methods have been widely explored for dynamic link prediction, as they can effectively capture network structure and evolution without requiring labeled data. In this section, we categorize these methods into two main types: random walk-based approaches and matrix analysis-based approaches.

Random walk-based methods. Random walk-based methods aim to learn low-dimensional representations of nodes and are widely applicable to dynamic networks. Zhang et al. [47] proposed a time-constrained random walk approach that avoids manually defined snapshot intervals and preserves temporal continuity. Nguyen et al. [48] introduced a general framework that incorporates temporal dependencies into random-walk-based models, effectively capturing time-ordered interactions while minimizing information loss. Causal Anonymous Walks (CAWs) [49] further enhance temporal modeling by using time-based random walks to extract temporal features, replacing node identities with node hitting counts, and establishing temporal correlations. These methods effectively capture temporal dynamics, though integrating rich node attribute information remains a significant challenge.

Matrix analysis-based methods. Matrix analysis-based methods predict links via matrix decomposition or completion. Non-negative Matrix Factorization (NMF) is widely used for learning latent structural features. Ahmed et al. [50] proposed an NMF-based method that employs novel iterative rules to construct matrix factors capturing key network features, thereby effectively representing the network dynamics. Ma et al. [51] established the equivalence between feature decomposition and NMF algorithms, providing a theoretical foundation for designing dynamic link prediction methods based on NMF. Building on this, they proposed a novel algorithm that factorizes each network snapshot, utilizes graph communicability to extract features, and then folds the resulting feature matrix to predict links. Lei et al. [52] proposed a novel method called Adaptive Multiple Non-negative Matrix Factorization (AM-NMF), which embeds dynamic networks into a low-dimensional space while preserving features across different snapshots. The method introduces an adaptive parameter that automatically adjusts the contribution of various terms and effectively fuses hidden information from multiple time slices. Ma et al. [53] proposed the Graph Regularized Nonnegative Matrix Factorization (GrNMF) algorithm, which decomposes the adjacency matrix of each time slice while simultaneously leveraging information from other slices as structural regularization to preserve both intra-slice structural features and inter-slice temporal continuity.

1.2.2 Supervised learning methods

Supervised methods learn from labeled data to predict the existence of links through classification or regression. Compared to unsupervised approaches, they often achieve higher predictive accuracy. These methods include both traditional machine learning models and deep learning frameworks.

Traditional machine learning methods. Methods like logistic regression, SVMs, and random forests treat link prediction as a binary classification task. They rely on manually extracted features such as node attributes and temporal patterns. Liu et al. [54] established time series features of dynamic networks based on network topology features and link generation time and combined statistical models with supervised learning methods to predict links on a weighted dynamic network. Singh et al. [55] proposed a PILHNB method for dynamic social network link prediction based on Latent Dirichlet Allocation (LDA) and Hidden Naive Bayes (HNB). It considers behavioral control elements such as relationship network structure, node attributes, node location information, and node popularity, and learns the changing patterns of these factors of the network. Currently, traditional machine learning methods are relatively mature, providing many options for constructing complete prediction models. However, they are sensitive to quality and quantity of training data.

Deep learning models. Deep learning methods, particularly Graph Neural Networks (GNNs), have achieved remarkable success in dynamic link prediction. GNNs are generally categorized into four types: Recurrent Graph Neural Networks (RecGNNs), Convolutional Graph Neural Networks (ConvGNNs), Graph Autoencoders (GAEs), and Spatio-Temporal Graph Neural Networks (STGNNs). In practice, a complete dynamic link prediction model often integrates multiple types of GNNs.

Jiao et al. [56] proposed a dynamic network embedding approach that learns low-dimensional node representations while preserving non-linear temporal patterns, using attention-enhanced RecGNNs to update embeddings with temporal dependencies. Pham et al. [57] introduced ComGCN, a hybrid model combining RecGNNs and ConvGNNs for dynamic link prediction. Kumar et al. [58] developed GNS, a simulator that models system dynamics through an encoder–processor–decoder architecture, demonstrating strong generalization capabilities. Zheng et al. [59] proposed a transition structure that adaptively personalizes node modeling and captures node dynamics by performing multi-step transition propagation and leveraging dual GNNs to extract informative neighborhood features. Chen et al. [60] proposed a deep learning model called Encoder–LSTM–Decoder (E-LSTM-D) for predicting dynamic links. This model can automatically learn structural and temporal features within a unified framework. Min et al. [61] proposed the Spatio-Temporal Graph Social Network (STGSN), which extracts structural features of dynamic social networks using ConvGNNs and captures temporal patterns through a temporal attention mechanism. Yu et al. [62] introduced DyGFormer, a Transformer-based method that models first-order historical interactions by encoding neighbor co-occurrence and segmenting node interaction sequences, enabling effective learning from long-term dependencies. Wen et al. [63] designed a framework called TREND, which integrates temporal events and dynamic nodes to jointly capture local and global features, and further incorporates the Hawkes process to model the interplay between temporal events.

Many methods have been developed to handle more complex scenarios, such as dynamic weighted and heterogeneous networks. Qin et al. [64] proposed IDEA, which combines error and scale difference minimization objectives to generate weighted snapshots, using a generator–discriminator structure within an encoder–decoder framework. Wei et al. [65] introduced NeiDyHNE for dynamic heterogeneous attributed networks, leveraging hierarchical structural and convolutional temporal attention modules to capture node semantics, neighborhood structures, and network evolution over time.

1.2.3 Summary of dynamic link prediction

Dynamic link prediction has seen substantial progress through both unsupervised and supervised learning paradigms. Unsupervised approaches, including random walk and matrix factorization methods, offer scalability and flexibility in learning from unlabeled data, while supervised techniques, ranging from classical machine learning to advanced deep learning models, demonstrate superior capability in capturing complex spatio-temporal dependencies. Recent research trends also address more sophisticated settings such as weighted, heterogeneous, and attributed dynamic networks. These developments suggest that the integration of multiple strategies, particularly hybrid models, is a growing trend in dynamic link prediction.

2 Traditional heuristic metrics

In this supplementary material, we present five representative heuristic methods for link prediction, which are used in the comparative experiments reported in the main paper. The methods considered are Common Neighbors [9], Adamic-Adar Index [11], Resource Allocation Index [13], Local Path Index [18], and Katz Index [14]. While these methods were originally designed for static link prediction, we adapt them to the dynamic link prediction setting. Specifically, we aggregate the first $l-1$ graph snapshots of each sample into a single static graph, and then apply the heuristic methods to this aggregated graph to compute the likelihood scores for potential links in the l-th snapshot. The details of these methods are as follows:

2.1 Common Neighbors (CN)

This index counts the number of common neighbors between two nodes u and v. The underlying assumption is that two nodes with more shared neighbors are more likely to be connected.

$$\mathrm{CN}(u,v) = |\Gamma(u) \cap \Gamma(v)|,$$

where $\Gamma(u)$ denotes the set of neighbors of node u.

2.2 Adamic-Adar Index (AA)

This metric refines the CN index by assigning lower weights to high-degree common neighbors, assuming that rare shared neighbors are more informative.

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

2.3 Resource Allocation Index (RA)

The RA index models a resource allocation process, where node u sends a unit resource to v through common neighbors. Common neighbors with higher degrees distribute their resource more thinly among their neighbors, resulting in less resource allocated to v.

$$\mathrm{RA}(u, v) = \sum_{w \in \Gamma(u) \cap \Gamma(v)} \frac{1}{|\Gamma(w)|}.$$

2.4 Local Path Index (LP)

The LP index extends the CN metric by incorporating paths of length 3, enhancing predictive power while maintaining computational efficiency.

$$\mathrm{LP}(u, v) = (A^2)_{uv} + \beta (A^3)_{uv},$$

where A is the adjacency matrix and β is a damping factor (typically $\beta = 0.01$).

2.5 Katz Index

The Katz index considers all paths between two nodes, giving exponentially smaller weights to longer paths.

$$\mathrm{Katz}(u, v) = \sum_{l=1}^{\infty} \beta^l \cdot (A^l)_{uv},$$

where β is a decay parameter and A^l denotes the number of paths of length l between u and v. To ensure convergence, β must be smaller than the reciprocal of the largest eigenvalue of A.

3 Dataset specification

In this section, we provide a detailed specification of the datasets used in the main paper, including the simulation tool, data generation process, and data partitioning.

3.1 Simulation tool

Due to the lack of real UANET datasets, we employ the freely available mobility scenario generation and analysis tool BonnMotion [66] to produce UAV mobility traces. BonnMotion is widely used in the simulation of MANETs, VANETs, and UANETs, and supports various classic mobility models, including the RW (Random Waypoint), MG (Manhattan Grid), RPG (Random Point Group), and GM (Gauss–Markov) models used in our work. The tool's command-line interface enables precise reproduction of experimental configurations, such as the number of nodes, flight speed, and simulation duration.

3.2 Dataset generation

The data generation process is as follows. First, based on the selected mobility model and input parameters, BonnMotion generates trajectory data for each node, capturing the temporal variations of node position, velocity, and direction. Next, it outputs a sequence of UAV location records in the format (node ID, timestamp, x-coordinate, y-coordinate). Using the coordinate information of all nodes, we generate the UAV network topology. For each timestamp, if the distance between two nodes is less than the communication radius, a link is established between them, and the link weight is calculated according to Eq. (1) in the main paper. This generates a snapshot sequence for the UANET, which relies solely on the UAV location information produced by BonnMotion.

3.3 Dataset partitioning

A sliding window of size 11 with a step size of 1 is applied over the snapshot sequence of length 80, yielding 70 samples. Among these, the first 50 samples are used as the training set, and the remaining 20 samples as the test set. For each sample, the first 10 snapshots serve as the historical graph data input to the model, and the last snapshot is used as the target graph for prediction. The generated datasets have been publicly released at https://github.com/turtlo6/MUST.

4 Supplementary comparison results

In the main paper, we compare the performance of MUST with state-of-the-art learning-based methods across varying sampling intervals and flight speeds, showing that MUST consistently outperforms them in terms of AUC

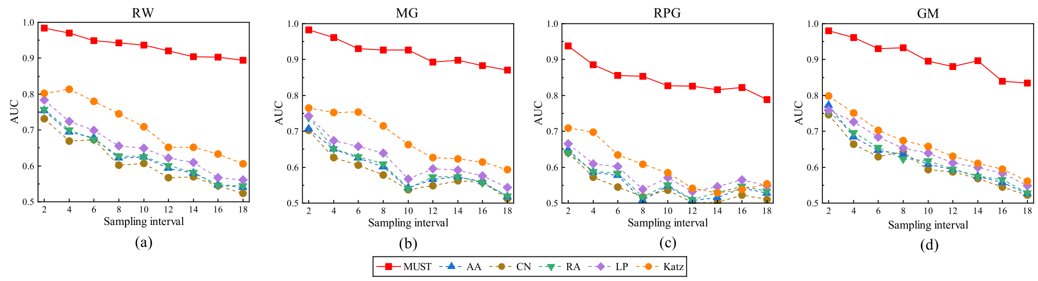

Figure 1: AUC vs. Sampling interval for MUST and traditional heuristic methods across the four mobility models.

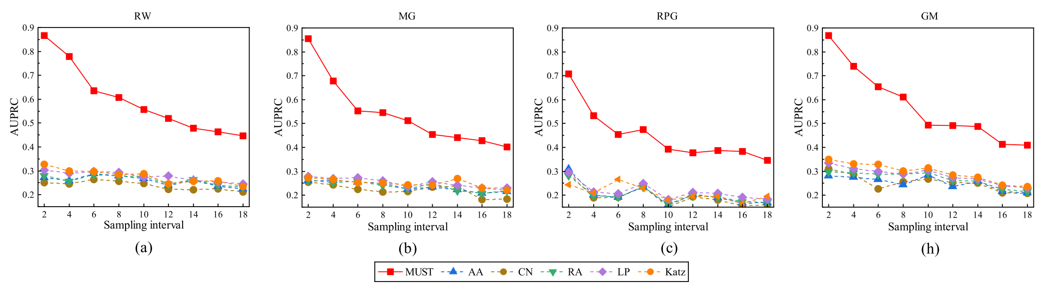

Figure 2: AUPRC vs. Sampling interval for MUST and traditional heuristic methods across the four mobility models.

and AUPRC. In this supplementary material, we further provide a comparison between MUST and representative heuristic methods, including CN, AA, RA, LP, and Katz, under varying sampling intervals and flight speeds, as a complement to the main paper. The experimental setup is identical to that described in the main paper.

4.1 Impact of snapshot sampling interval

Figs. 1 and 2 illustrate the AUC and AUPRC results under different sampling intervals, respectively. We observe that both MUST and the heuristic methods exhibit a decline in performance as the sampling interval increases. This is because larger sampling intervals reduce the temporal correlation between consecutive snapshots, thereby increasing the difficulty of link prediction. Among the heuristic methods, Katz achieves the best overall performance, while CN performs the worst. The superior performance of Katz

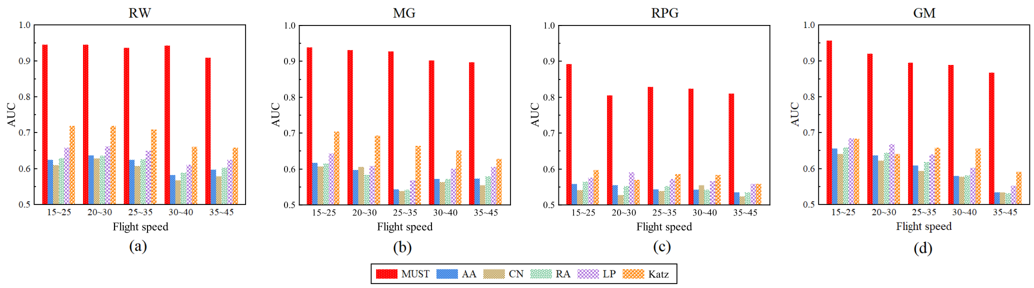

Figure 3: AUC vs. Flight speed for MUST and traditional heuristic methods across the four mobility models.

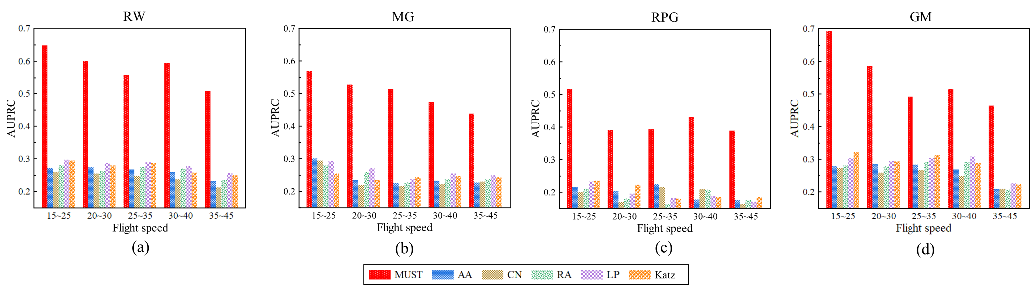

Figure 4: AUPRC vs. Flight speed for MUST and traditional heuristic methods across the four mobility models.

can be attributed to its ability to aggregate contributions from paths of all lengths, providing a more comprehensive characterization of node associations, whereas CN relies solely on direct connections among local neighbors and ignores information from longer paths between node pairs. Our method, MUST, significantly outperforms all heuristic methods across all sampling intervals, as heuristics are inherently based on handcrafted static topological rules and thus fail to effectively capture the evolving patterns of UANET topology.

4.2 Impact of flight speed

Figs. 3 and 4 present the performance of MUST and the heuristic methods under different flight speeds. As the flight speed increases, both MUST and the heuristic methods exhibit a decline with fluctuations in AUC and AUPRC. This is because higher flight speeds induce more rapid topological changes, thereby making link prediction more challenging. Among the heuristic methods, Katz once again achieves the best overall performance, whereas CN performs the worst. In contrast, MUST consistently outperforms all heuristic methods across different flight speeds.

References

[1] D. Arrar, N. Kamel, and A. Lakhfif, "A comprehensive survey of link prediction methods: D. arrar et al.," *The journal of supercomputing*, vol. 80, no. 3, pp. 3902–3942, 2024.

[2] S. U. Balvir, M. M. Raghuwanshi, and K. R. Singh, "A comprehensive survey on learning based methods for link prediction problem," in *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*, pp. 1–7, IEEE, 2023.

[3] A. Divakaran and A. Mohan, "Temporal link prediction: A survey," *New Generation Computing*, vol. 38, no. 1, pp. 213–258, 2020.

[4] A. Kumar, S. S. Singh, K. Singh, and B. Biswas, "Link prediction techniques, applications, and performance: A survey," *Physica A: Statistical Mechanics and its Applications*, vol. 553, p. 124289, 2020.

[5] 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.

[6] M. Qin and D.-Y. Yeung, "Temporal link prediction: A unified framework, taxonomy, and review," *ACM Computing Surveys*, vol. 56, no. 4, pp. 1–40, 2023.

[7] H. Wu, C. Song, Y. Ge, and T. Ge, "Link prediction on complex networks: an experimental survey," *Data science and engineering*, vol. 7, no. 3, pp. 253–278, 2022.

[8] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proceedings of the twelfth international conference on Information and knowledge management*, pp. 556–559, 2003.

[9] M. Newman, "Clustering and preferential attachment in growing networks," *Research Papers in Economics,Research Papers in Economics*, Mar 2001.

[10] A. Barabási, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, and T. Vicsek, "Evolution of the social network of scientific collaborations," *Physica A: Statistical Mechanics and its Applications*, p. 590–614, Aug 2002.

[11] E. Adar and L. Adamic, "Friends and neighbors on the web," *First Monday,First Monday*, Jan 2001.

[12] P. Jaccard, "Distribution de la flore alpine dans le bassin des dranses et dans quelques régions voisines,"

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

[14] L. Katz, "A new status index derived from sociometric analysis," *Psychometrika*, p. 39–43.

[15] H. Tong, C. Faloutsos, and J.-y. Pan, "Fast random walk with restart and its applications," in *Sixth International Conference on Data Mining (ICDM'06)*, Dec 2006.

[16] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proceedings of the twelfth international conference on Information and knowledge management*, Nov 2003.

[17] Z. Xu, C. Pu, and J. Yang, "Link prediction based on path entropy," *Physica A: Statistical Mechanics and its Applications*, p. 294–301, Aug 2016.

[18] L. Lü, C.-H. Jin, and T. Zhou, "Similarity index based on local paths for link prediction of complex networks," *Physical Review E*, Nov 2009.

[19] W. Liu and L. Lü, "Link prediction based on local random walk," *EPL (Europhysics Letters)*, p. 58007, Mar 2010.

[20] X. Wang, X. Zhang, C. Zhao, Z. Xie, S. Zhang, and D. Yi, "Predicting link directions using local directed path," *Physica A: Statistical Mechanics and its Applications*, vol. 419, p. 260–267, Feb 2015.

[21] C. Wang, V. Satuluri, and S. Parthasarathy, "Local probabilistic models for link prediction," in *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, Oct 2007.

[22] L. Getoor, N. Friedman, D. Koller, and B. Taskar, "Learning probabilistic models of link structure," *Journal of Machine Learning Research,Journal of Machine Learning Research*, Mar 2003.

[23] D. Jensen and J. Neville, "Statistical models and analysis techniques for learning in relational data," Jan 2006.

[24] B. Taskar, W.-f. Patrick., P. Abbeel, and D. Koller, "Link prediction in relational data," *Neural Information Processing Systems,Neural Information Processing Systems*, Dec 2003.

[25] A. Clauset, C. Moore, and M. E. J. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, p. 98–101, May 2008.

[26] H. C. White, S. A. Boorman, and R. L. Breiger, "Social structure from multiple networks. i. blockmodels of roles and positions," *American Journal of Sociology*, p. 730–780, Jan 1976.

[27] P. W. Holland, K. B. Laskey, and S. Leinhardt, "Stochastic blockmodels: First steps," *Social Networks*, p. 109–137, Jun 1983.

[28] R. Guimerà and M. Sales-Pardo, "Missing and spurious interactions and the reconstruction of complex networks," *Proceedings of the National Academy of Sciences*, p. 22073–22078, Dec 2009.

[29] P. W. Holland and S. Leinhardt, "An exponential family of probability distributions for directed graphs," *Journal of the American Statistical Association*, vol. 76, p. 33–50, Mar 1981.

[30] J. Park and M. Newman, "Solution of the two-star model of a network," *Physical Review E,Physical Review E*, Dec 2004.

[31] L. Pan, T. Zhou, L. Lü, and C.-K. Hu, "Predicting missing links and identifying spurious links via likelihood analysis," *Scientific Reports*, vol. 6, Mar 2016.

[32] S. Wasserman and P. Pattison, "Logit models and logistic regressions for social networks: I. an introduction to markov graphs andp," *Psychometrika*, p. 401–425, Sep 1996.

[33] M. Belkin and P. Niyogi, *Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering*. Jan 2002.

[34] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, p. 2323–2326, Dec 2000.

[35] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, p. 2319–2323, Dec 2000.

[36] B. Perozzi and S. Skiena, "Deepwalk: Online learning of social representations,"

[37] B. Krishnapuram, M. Shah, A. Smola, C. Aggarwal, D. Shen, and R. Rastogi, "Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining," Aug 2016.

[38] A. K. Menon and C. Elkan, *Link Prediction via Matrix Factorization*, p. 437–452. Jan 2011.

[39] B. Chen, F. Li, S. Chen, R. Hu, and L. Chen, "Link prediction based on non-negative matrix factorization," *PLOS ONE*, vol. 12, p. e0182968, Aug 2017.

[40] M. Hasan, V. Chaoji, S. Salem, and M. Zaki, "Link prediction using supervised learning," Jan 2006.

[41] H. Kashima and N. Abe, "A parameterized probabilistic model of network evolution for supervised link prediction," in *Sixth International Conference on Data Mining (ICDM'06)*, p. 340–349, Dec 2006.

[42] M. A. Hasan and M. J. Zaki, *A Survey of Link Prediction in Social Networks*, p. 243–275. Jan 2011.

[43] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *European semantic web conference*, pp. 593–607, Springer, 2018.

[44] T. Kipf and M. Welling, "Variational graph auto-encoders," *arXiv: Machine Learning,arXiv: Machine Learning*, Nov 2016.

[45] 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*, KDD '17, (New York, NY, USA), p. 575–583, Association for Computing Machinery, 2017.

[46] M. Zhang and Y. Chen, "Link prediction based on graph neural networks," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, (Red Hook, NY, USA), p. 5171–5181, Curran Associates Inc., 2018.

[47] M. Zhang, B. Xu, and L. Wang, "Dynamic network link prediction based on random walking and time aggregation,"

[48] G. H. Nguyen, J. Boaz Lee, R. A. Rossi, N. K. Ahmed, E. Koh, and S. Kim, "Dynamic network embeddings: From random walks to temporal random walks," in *2018 IEEE International Conference on Big Data (Big Data)*, pp. 1085–1092, 2018.

[49] Y. Wang, Y. Chang, Y. Liu, J. Leskovec, and L. Pan, "Inductive representation learning in temporal networks via causal anonymous walks," *Learning,Learning*, Jan 2021.

[50] N. M. Ahmed, L. Chen, Y. Wang, B. Li, Y. Li, and W. Liu, "Deepeye: Link prediction in dynamic networks based on non-negative matrix factorization," *Big Data Mining and Analytics*, vol. 1, no. 1, pp. 19–33, 2018.

[51] X. Ma, P. Sun, and G. Qin, "Nonnegative matrix factorization algorithms for link prediction in temporal networks using graph communicability," *Pattern Recognition*, p. 361–374, Nov 2017.

[52] K. Lei, M. Qin, B. Bai, and G. Zhang, "Adaptive multiple non-negative matrix factorization for temporal link prediction in dynamic networks," in *Proceedings of the 2018 workshop on network meets AI & ML*, pp. 28–34, 2018.

[53] X. Ma, P. Sun, and Y. Wang, "Graph regularized nonnegative matrix factorization for temporal link prediction in dynamic networks," *Physica A: Statistical Mechanics and its Applications*, p. 121–136, Apr 2018.

[54] J. Liu, Y. Jiang, Y. Wang, H. Xie, and J. Ni, "Link prediction in dynamic networks based on machine learning," in *2020 3rd International Conference on Unmanned Systems (ICUS)*, p. 836–841, Nov 2020.

[55] A. K. Singh and K. Lakshmanan, "Pilhnb: Popularity, interests, location used hidden naive bayesian-based model for link prediction in dynamic social networks," *Neurocomputing*, vol. 461, pp. 562–576, 2021.

[56] P. Jiao, X. Guo, X. Jing, D. He, H. Wu, S. Pan, M. Gong, and W. Wang, "Temporal network embedding for link prediction via vae joint attention mechanism," *IEEE Transactions on Neural Networks and Learning Systems*, p. 7400–7413, Dec 2022.

[57] P. Pham, L. T. T. Nguyen, N. T. Nguyen, W. Pedrycz, U. Yun, and B. Vo, "Comgcn: Community-driven graph convolutional network for link prediction in dynamic networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, p. 5481–5493, Sep 2022.

[58] K. Kumar and J. Vantassel, "Gns: A generalizable graph neural network-based simulator for particulate and fluid modeling," Nov 2022.

[59] T. Zheng, Z. Feng, T. Zhang, Y. Hao, M. Song, X. Wang, X. Wang, J. Zhao, and C. Chen, "Transition propagation graph neural networks for temporal networks," *IEEE Transactions on Neural Networks and Learning Systems*, p. 1–13, Jan 2022.

[60] J. Chen, J. Zhang, X. Xu, F. Cheng-bo, D. Zhang, Q. Zhang, and Q. Xuan, "E-lstm-d: A deep learning framework for dynamic network link prediction," *Cornell University - arXiv,Cornell University - arXiv*, Feb 2019.

[61] S. Min, Z. Gao, J. Peng, L. Wang, K. Qin, and B. Fang, "Stgsn — a spatial–temporal graph neural network framework for time-evolving social networks," *Knowledge-Based Systems*, p. 106746, Feb 2021.

[62] L. Yu, L. Sun, B. Du, and W. Lv, "Towards better dynamic graph learning: New architecture and unified library," Mar 2023.

[63] Z. Wen and Y. Fang, "Trend: Temporal event and node dynamics for graph representation learning,"

[64] M. Qin, C. Zhang, B. Bai, G. Zhang, and D.-Y. Yeung, "High-quality temporal link prediction for weighted dynamic graphs via inductive embedding aggregation," *IEEE Transactions on Knowledge and Data Engineering*, p. 9378–9393, Sep 2023.

[65] X. Wei, W. Wang, C. Zhang, W. Ding, B. Wang, Y. Qian, Z. Han, and C. Su, "Neighbor-enhanced representation learning for link prediction in dynamic heterogeneous attributed networks," *ACM Transactions on Knowledge Discovery from Data*, vol. 18, no. 8, pp. 1–25, 2024.

[66] N. Aschenbruck, R. Ernst, E. Gerhards-Padilla, and M. Schwamborn, "Bonnmotion: a mobility scenario generation and analysis tool," in *Proceedings of the 3rd international ICST conference on simulation tools and techniques*, pp. 1–10, 2010.