Chapter 2

Preliminaries

This chapter gives a brief introduction to the literature and standard techniques of link prediction. Also, information about the evaluation of link prediction algorithms is presented. These evaluation procedures are followed for the proposed works of this thesis. At last, the background of the experiments conducted in this thesis is presented, divided into the categories of link prediction in simple and multiplex networks.

2.1 Survey of Elementary Link Prediction Techniques

Different categories of methods have been proposed for link prediction that can be broadly classified as similarity-based [12], probabilistic and maximum likelihood-based [14, 15], and dimensionality reduction-based techniques. The similarity-based link prediction category can further be separated into local similarity-based methods, global similarity-based methods, and quasi-local similarity-based methods. A brief introduction to these various categories of link prediction algorithms is presented in the following sections. This thesis's proposed research primarily focuses on similarity-based link prediction.

2.1.1 Similarity-based Link Prediction

Similarity-based link prediction is computationally the most straightforward kind of link prediction category, in which similarity scores are calculated for node pairs that have to be evaluated for the possibility of link existence [33, 34]. These methods can use information from either the immediate neighborhood of nodes themselves (local similarity-based) or take the entire graph structure into account (global similarity-based). New methods containing trade-offs between these separate categories of information have also recently emerged (quasi-local similarity-based), showing relative improvements in the overall task of link prediction.

2.1.1.1 Local Similarity-based Link Prediction

Local similarity-based features are calculated using information from immediate neighbors. Neighbors are the closest ones to a given user. The following are examples of link prediction methods of this category -

- **Common Neighbors (CN) [16].** The size of common neighbours for a given pair of nodes x and y is determined as the intersection of the two node neighbourhoods in a particular network or graph. It is calculated as.

$$S(x,y) = |\Gamma(x) \cap \Gamma(y)|, \tag{2.1}$$

  where $\Gamma(x) \& \Gamma(y)$ - neighbors of the node x and y respectively.

- **Jaccard Coefficient(JC) [19]**. It normalizes the size of common neighbour. It is calculates as.

$$S(x,y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}. \tag{2.2}$$

- **Adamic/Adar Index(AA)** [17]. Adamic and Adar presented a metric that uses shared features to produce a similarity score between two web pages, which is then used in link prediction. Mathematically, it is expressed as.

$$S(x,y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log k_z}, \tag{2.3}$$

where k_z is the degree of the node z.

- **Preferential Attachment (PA) [16].** According to the Preferential Attachment metric, new links are more likely to contact higher-degree nodes than lower ones. The likelihood increment new connection associated with a node x is proportional to k_x, the degree of the node. The following formula is used to estimate the PA score between two nodes x and y.

$$S(x,y) = k_x.k_y. \tag{2.4}$$

- **Resource Allocation (RA) [18].** Resource Allocation index is based on the hypothetical where we suppose node x sends some resources to y through the common nodes of both x and y then the similarity between the two vertices is computed in terms of resources sent from x to y.

$$S(x,y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{k_z}. \tag{2.5}$$

2.1.1.2 Global Similarity-based Link Prediction

Global similarity-based features are usually calculated using information from the whole graph structure. The following are examples of link prediction methods of this category -

- **Shortest Path (SP) [35].** There are several different algorithms that can be used to determine the shortest path between two vertex pairs in a graph depending on circumstances [36–38]. Shortest Path is calculated as.

$$SP(x,y) = -|d(x,y)|, \tag{2.6}$$

where the shortest path $d(x,y)$ between the node pair (x,y) is calculated using the Dijkstra algorithm [36].

- **Cos+ (COSP) [39].** Any inner product metric, such as the cosine similarity, can be used to determine how similar two nodes x and y are to one another. The cosine similarity time metric is based on $L^{\dagger}$ by calculating similarity of two vectors. It is calculated by the following formula.

$$COSP(x,y) = \frac{L^{\dagger}_{x,y}}{\sqrt{L^{\dagger}_{x,x}L^{\dagger}_{y,y}}}. \tag{2.7}$$

- **Matrix Forest Index (MFI) [40].** MFI employs the spanning tree principle. It contains fewer links than the original graph.

$$MFI = (I+L)^{-1}, \tag{2.8}$$

where $(I+L)_{(x,y)}$ is the count of spanning rooted forests (x as root) that include both the nodes x and y. This value is also identical to the co-factor of $(I+L)_{(x,y)}$.

- **Average Commute Time (ACT) [41].** To calculate the average commute time, the random walk method is employed.

$$n(x,y) = m(x,y) + m(y,x). \tag{2.9}$$

This above equation can be made simpler using the pseudo-inverse of the Laplacian matrix L^{+}

$$n(x,y) = |E|(l^{+}_{xx} + l^{+}_{yy} - 2l^{+}_{xy}), \tag{2.10}$$

where l^{+}_{xy} denotes the (x,y) entry of the matrix L^{+}. Pseudo-inverse of the Laplacian, L^{+} can be computed as

$$L^{+} = (L - \frac{ee^{T}}{n})^{-1} + \frac{ee^{T}}{n}, \tag{2.11}$$

where e denotes a column vector consisting of 1's.

- **Katz index (KATZ) [22].** A variation of the shortest route metric is the KATZ index. In order to punish longer trips, it dumps exponentially for all direct pathways between x and y. Here, $paths_{x,y}^{<l>}$ is considered as the set of total l length paths between x and y, β is a damping factor that controls the path weights and A is the adjacency matrix.

$$S(x,y) = \sum_{l=1}^{\infty} \beta^{l} |paths_{x,y}^{<l>}| = \sum_{l=1}^{\infty} \beta^{l} (A^{l})_{x,y}, \tag{2.12}$$

2.1.1.3 Quasi-Local Similarity-based Link Prediction

Quasi-local similarity-based features are usually calculated using a calculated trade-off between local and global information such that both accuracy and efficiency are maintained up to an acceptable level. The following are examples of link prediction methods of this category -

- **Local Path Index (LPI) [26].** The local paths of lengths 2 and lengths 3 are used to calculate the LPI metric. It utilizes some additional information from the neighbors within a length 3 distance from the current node, in contrast to metrics that only use the information of the nearest neighbors. It is calculated as.

$$S^{LP} = A^{2} + \varepsilon A^{3}, \tag{2.13}$$

where ε is free parameter. It is obvious that the measurement converges to the common neighbour when $\varepsilon = 0$. If x and y are not directly connected, $(A^{3})_{xy}$ is equated to the total different paths of length 3 between x and y. The index may also be extended to take on a generalized form.

$$S^{LP} = A^{2} + \varepsilon A^{3} + \varepsilon^{2} A^{4} + \ldots + \varepsilon^{(n-2)} A^{n}, \tag{2.14}$$

where n is the maximal order. Computing this index becomes more complicated with the increasing value of n.

- **Path of Length 3 (L3) [27].** The path length is mathematically, calculated by the following equation.

$$S(x,y) = \sum_{u,v} \frac{a_{x,u}.a_{u,v}.a_{v,y}}{\sqrt{k_u.k_v}}, \tag{2.15}$$

where $a_{x,u}$ denotes the interaction strength among nodes x and u and k_u denotes the degree of node u. The Eq. 2.15 measures the connectivity among node u and node v by utilizing the degree-normalized adjacency matrix.

- **Clustering Coefficient based Link Prediction (*CCLP*) [42].** In this method, the shared neighbors of the seed node pair are taken into account for calculating the similarity score. To determine the similarity score, the clustering coefficient of these common neighbors is used. The following formula is used to calculate the similarity score between two disconnected seed node pairs.

$$CCLP(a,b) = \sum_{c \in \Gamma(A) \cap \Gamma(B)} C(c), \tag{2.16}$$

where C(c)= $\frac{t(c)}{k_c(k_c-1)}$ and k_c is the degree of the node c, CN(a,c) is the number of common neighbors of the nodes a and c, C(c) is the clustering coefficient of the node c.

- **Node and Link Clustering Coefficient (NLC) [25].** The ”Clustering Coefficient”, a fundamental aspect of a network's topology, serves as the foundation for this similarity measure. The clustering coefficients of both nodes and links are incorporated to compute the similarity score.

$$S(x,y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{|\Gamma(x) \cap \Gamma(z)|}{k_z - 1} \times C(z) + \frac{|\Gamma(y) \cap \Gamma(z)|}{k_z - 1} \times C(z). \tag{2.17}$$

- **CAR-based Common Neighbor Index (*CAR*) [20].** On the premise that the likelihood of a connection existing between two nodes increases if their shared neighbours are also members of the same local community, CAR-based indices have been proposed. Here, $LCL(x,y)$ refers to the number of local community links which are defined as the links among the common neighbors of seed nodes x and y. $\gamma(z)$ is the subset of neighbors of node z that are also common neighbors of x and y.

$$S(x,y) = CN(x,y) \times LCL(x,y) = CN(x,y) \times \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{|\gamma(z)|}{2}, \tag{2.18}$$

- **Local Naive Bayes-based Common Neighbors (*LNBCN*) [21].** The LNBCN technique is based on the Naive Bayes theory and the claims that numerous different neighbours perform various roles in the network and, as a result, contribute differently to the score function generated for unobserved node pairings. Here, $C(z)$ is node clustering coefficient and ρ is the network density.

$$S(x,y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} [\log(\frac{C(z)}{1-C(z)}) + \log(\frac{1-\rho}{\rho})], \tag{2.19}$$

2.1.2 Probabilistic and Maximum Likelihood-based Link Prediction

For any given graph, probabilistic models evaluate a given objective function and optimize it based on different parameters of the graph [14, 43–45]. A new link's probability is estimated using conditional probability, which estimates its presence on the overall graph-based objective function. A link that enhances the parameters of a graph, such as overall connectivity, is given more importance.

2.1.3 Dimensionality Reduction-based Link Prediction

Link prediction techniques based on the frameworks of network embedding and matrix decomposition come under the subset of dimensionality reduction [46–52]. This is because, at their core, these techniques transform the whole graph, which contains several nodes and edges, into fixed-length edge and node vectors which are further distilled to calculate link likelihoods. Embedding-based techniques extract node vectors from the original graph such that similar nodes with approximately matching neighborhoods have small-dimension vector representations very close to each other. The embedding space for such tasks is much smaller than the original unrefined representation of graphs such as those in the adjacency matrix form. Another class of dimensionality reduction techniques is matrix factorization/decomposition-based techniques.

2.2 Evaluation of Link Prediction Algorithm Performance

2.2.1 Performance Evaluation

The essential task of any link prediction algorithm is to generate link likelihoods or probabilities for edges not present in the original graph. In the case of single-layered graph link prediction, these probabilities are based on the topological properties of the graph. In contrast, layer-specific link prediction in a multiplex network is based on the combined information provided by all graph layers. The performance of link prediction algorithms can be evaluated using different methods. In this thesis, all algorithms are evaluated by discerning the probabilities of all possible edges and then comparing these probabilities and corresponding labels with the edges of the test graph. Initially, the datasets are randomly divided into test and training graphs from the entire set of edges.

The ratio of the number of edges in the testing graph to the total number of edges is represented by the *Ratio* variable. Other strategies, such as sampling an equal number of non-edges with test edges, also exist to evaluate link prediction algorithms. These strategies help simplify the overall experiment by limiting the number of predictions made for a smaller candidate edge set. Another link prediction algorithm evaluation type involves generating edge probabilities for all possible edges but using only a certain percentage of the highest probabilities for performance evaluations, i.e., top-*k* edges prediction. However, we have not used such algorithm evaluations in this thesis to exhaustively test our proposals for providing a solution to the unbalanced classification-based link prediction problem.

2.2.2 Evaluation Metrics

Hasan et al.[53] structured the link prediction issue as a binary classification problem, allowing for the application of the majority of associated evaluation metrics. A confusion matrix may be used to illustrate the assessment of a binary classification issue with two classes [54]. Three measures are used to assess the proposed method *ELP*: accuracy, area under the precision-recall curve (AUPR), and area under the curve (AUC).

- **Accuracy -** Accuracy score is the simplest kind of measure that may be used to evaluate the effectiveness of a classification system. It is just the sum of all accurate predictions divided by the whole sample size. The link prediction problem's valid predictions may be specified using the link existence probability. If the likelihood is greater than 0.5 and an actual connection exists in the graph, the classification is regarded accurate.
- **AUPR -** For binary classification issues, AUPR is more informative and beneficial [54, 55]. As a result, it is employed as a forecasting measure. The AUPR values are determined using the precision-recall curve, in which the x- and y-axes reflect the recall and precision values, respectively.

- **AUC -** The area under the receiver operating characteristics curve (AUROC/AUC) [54, 55] plots TPR (y-axis) against FPR (x-axis). The AUC value is a single-point statistical summary with a range of 0–1 and is estimated using the trapezoidal rule.
- **F1 -** The F1 score employs as a comparative measurement metric for two classifiers where one classifier was good in precision while the other was good in recall values. In such cases, the F1 score predicts the better classifier between them by taking the harmonic means of precision and recall values. A higher value of the F1 score for an algorithm represents its better performance compared to others and the absolute values range between 0&1.

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{2.20}$$

2.3 Link Prediction in Simple Networks

2.3.1 Recent Related Work

Numerous comprehensive studies have been done in order to provide a comprehensive analysis of the link prediction problem and its associated literature [43, 44, 56–60]. These studies have classified link prediction into several classes like similarity-based [17, 18], dimensionality reduction-based [61, 62], probabilistic and maximum likelihood-based [63, 64], learning-based [65], information theory-based [66], etc. The probabilistic link prediction model optimizes an objective function for a particular network in order to construct a model made of several parameters. This model does an excellent job of estimating observed data for the given network. At that point, the probability of a non-existing connection being present is defined as the value of the objective function in the presence of such a link. Techniques for dimensionality reduction fall into two categories: embedding-based and matrix-decomposition-based. This method of link prediction generates feature sets for relevant edges and trains machine learning classifiers on them for classification tasks. While learning-based strategies also make use of machine learning classifiers, the features in this issue category correspond to various attributes of edges on various link prediction indices. Similarity-based link prediction methods are most popular in practice, particularly structural similarities due to their simplicity and efficiency. These methods estimate the similarity index for a pair of individuals based on local, quasi-local, and global topological information.

A different form of solution to the link prediction problem is via graph embedding techniques [61, 62, 67–69], which can be used in binary classification problems in combination with machine learning algorithms. Logically linear embedding (LLE) [61] and Laplacian eigenmaps [67] are some of the matrix-based graph embedding methods that can be used. However, considering their complicated overall implementation and greater resource intensiveness, they are not scalable. For addressing the scalability issue on large graphs, the sparsity of networks can be used as an improvement factor in targeted algorithms. To deal with the limitations of complete matrix-based embedding methods, DeepWalk, a local embedding-based method that uses local information of random walk, was proposed by Perozzi et al.[62]. DeepWalk preserves higher-order proximity by maximizing the probability of co-occurrence of random walk. The authors of [68] also use a directed random walk model to embed the nodes using a corpus of 2 hop possibilities. For performance improvement, the authors of this node2vec algorithm also incorporated the concept of combining both depth-first and breadth-first searches in possible paths. Random walk-based link prediction methods are also considered quasi-local link prediction methods, such as ones by Berahmand et al.[70, 71].

The newest subset of link prediction algorithms is community-guided link prediction. A topology-based link prediction algorithm was proposed by Huang Zan[72]. Using a cycle formation model, the generalized clustering coefficient was used as the likelihood score. In 2015, a resolution-based community division-based link prediction approach was presented by Ding et al.[73]. They proposed to use the coarsened resolution to extract community structure in the first step. For computing target link probability, a frequency statistical model is used to distinguish different communities. In 2016, a new similarity feature called community relevance was proposed by Ding et al.[74]. They proposed an amalgamated feature which, in addition to other topological information usually used in other classical link prediction approaches, also uses latent cross-community information. Another recent algorithm was CLP-ID, proposed by Singh et al.[75], that combines a community-based framework and information diffusion principle to calculate the prediction scores for target links. Community detection approaches which use deep learning are also being researched in current times [76–78]. Motif based link prediction has been proposed by Rossi et al.[79]. Multiple similarity based link prediction algorithms were combined with stacked machine learning algorithms to produce improved performance on small datasets by Li et al.[80]. Bastami et al.[81] proposed a gravitation inspired method which combines local, global and community-based features to improve upon the performance of similarity-based methods. In case of signed network link prediction becomes a three class prediction problem but methods which work on unsigned networks have also been successfully used with modification for signed link prediction (Chen et al.[82]). Similarly, Liu et al.[83] used motifs for signed link prediction in complex networks. Due to the advent of cloud computing, link prediction algorithms which are designed specifically for parallel computing platforms are also a growing area of interest. One such proposal was made by Wang et al.[84] which provides community enhanced similarity-based scores to provide an algorithm which can be parallelized across many clusters.

2.3.2 Experimental Setup

All of the experiments performed on a 64-bit Linux Mint 19.3 PC with Intel(R) Core(TM) i7-4770 CPU@ 3.40GHz processor and 32GB memory. Python was used as language of programming all the algorithms. The code for link prediction on specific layers is available on Github at https://github.com/shivansh-mishra/linkpredict-static.

TABLE 2.1: Statistical information of real-world datasets

Dataset	N	E	D	K	C
Karate	34	78	2.34	4.59	0.57
Jazz	198	2742	2.224	27.697	0.617
Airlines	235	1297	2.31	11.04	0.558
Celegansneural	297	2148	2.45	14.47	0.29
Political blogs	1490	16718	2.738	22.44	0.361
SmaGri	1059	4917	2.981	9.286	0.349
GrQc	5242	14496	6.047	5.531	0.53

Each experiment was executed for each algorithm and testing edges to total edges percent value ($Ratio$) 20 times.

2.3.3 Datasets

To validate the proposed algorithms in this thesis, seven real-world simple one-layered networks are used to compare their performance with other methods. Karate[1] [85] dataset contains social ties among the members of a university karate club collected by Wayne Zachary in 1977. Jazz[2] [86] is a collaboration network of jazz musicians which shows musicians as nodes and edges represent their respective collaborations with each other. Airlines[3] [87] is a US airline network where nodes and edges represent airports and the connectivity between airports. Celegansneural[4] [88] is a neural network of C. Elegans. Nodes represent neurons, and edges denote connections by either a synapse or a gap junction. Political blogs[5] [17] is a network of hyperlinks between weblogs on US politics,

[1] http://vlado.fmf.uni-lj.si/pub/networks/data/ucinet/ucidata.htm

[2] http://networkrepository.com/jazz.php

[3] http://vlado.fmf.uni-lj.si/pub/networks/data/

[4] https://neurodata.io/project/connectomes/

[5] http://www-personal.umich.edu/ mejn/netdata/

recorded in 2005 by Adamic and Glance. SmaGri[6] [89] is the undirected form of a citation network from Garfield collection which represents the results searches in Web of Science. It was made using HistCite software. Table 2.1 provides the statistical information about datasets used for experimental analysis. N, E, D, K, and C represent the number of nodes, number of edges, average shortest distance between a pair of node, the average degree of node, and average clustering coefficient of the network respectively.

2.3.4 Baseline Methods

The following methods are used as baseline algorithms for the comparison on link prediction performance on simple networks. Newman et al.[16] stated that the similarity score between a pair of nodes is dependent on the number of common neighbors between them and called the method Common Neighbor (*CN*). Zhou et al.[18] proposed the *RA* index for link prediction using diffusion model to penalize higher-degree nodes and called it Resource Allocation (*RA*). Cannistraci et al.[20] suggested a similarity score using the local community paradigm. They proposed *CAR* variants of *CN*, *JA*, *AA*, and *RA* of which the *CN* variant is used for this research. Wu et al.[42] used the clustering coefficient to get a better understanding of the strength of a possible link and proposed Clustering coefficient link prediction (*CCLP*). Jaccard Coefficient (*JC*) is one of the oldest metric proposed by Jaccard[19] and is a bit similar to Common Neighbor based similarity score. Barabasi et al.[90] considered an opposite approach to the diffusion paradigm specifically for co-authorship networks and called it Preferential attachment (*PA*). *PageRank* is an adjustment of *Katz* centrality that takes into consideration the fact that the centrality gain using a link from an important node should be decreased depending on how many nodes the central node is connected to (resource allocation). Grover et al.[68] presented a network embedding based method for link prediction. $N2V(Node2vec)$ creates embeddings for nodes in a low dimensional space and then formulates edge embeddings from these node embeddings. These are used as

[6] http://vlado.fmf.uni-lj.si/pub/networks/data/cite/default.htm

training data for a logistic regression based classification model. Singh et al.[75] proposed an algorithm, $CLP-ID$, which combines a community-based framework and information diffusion to calculate the prediction scores for target links.

2.4 Link Prediction in Multiplex Networks

2.4.1 Recent Related Work

In the last decade, many link prediction techniques have been used to predict missing links in static networks. Recently, the focus has shifted to more complex networks like heterogeneous, multilayer, and multiplex networks. This section presents the related works on link prediction in multiplex networks. In order to extend the concept of extending static network based similarity indices to the multiplex setting, some fundamental issues have to be dealt with [91]. These challenges include different node behaviour in different networks layers along with the degree to which different layers contribute to each others' structure. Recently there has been a lot of research interest in the field of link prediction in multiplex networks such as ones by Shakibian et al.[92], De Bacco et al.[93], Koptelov et al.[94] and Fan et al.[95].

Multiplex networks are heterogeneous networks that use a variety of connection types yet share the same nodes [96]. To illustrate such a structure, we may use the notion of stacked layers of graphs (all with the same node set and 2D representation). Together, these connected layers form the whole multiplex network [97–99]. The task of link prediction in one specific layer makes use of connectivity information of nodes in all other layers as well. In Hristova et al.[100], authors demonstrate the connection between interaction frequency of users and network multiplexity, making use of both social and spatial information. To anticipate links, feature sets based on edges are produced and passed into a machine learning classifier. Jalili et al.[101] make similar predictions using node and path-based characteristics. Sharma and Singh[102] tweaked this technique to link prediction based on the presence of comparable edges in other layers. Pujari and Kanawati[103] suggested a similar supervised learning-based link prediction technique, but theirs makes use of similarities measured across all layers concurrently. Hajibagheri et al.[104] suggested an approach in which, rather than computing edge-based similarities across all layers, inter-layer similarities are utilised to reweight the original layer's link prediction score. They aggregate several topological similarities into a single result via rank based estimations.

Yao et al.[105] proposed a method which combines both inter and intra-layer information for more accurate link prediction. Mandal et al.[106] created a multiplex network dataset using edge information from both Twitter and Foursquare social networking platforms, and performed link prediction using supervised machine learning based algorithms. Najari et al.[107] used Logistic regression on feature sets based on topological information of all layers to predict edges. While others used link prediction in multiplex network as a method of providing a more complete multiplex graph, Samei and Jalili[108] proposed a method which identifies spurious links which represent noise in the network. The same authors proposed two more similarity-based indices based on hyperbolic distances between nodes. These indices were used to identify both new and spurious links in the network hence converting the binary link prediction problem to a three classes-based classification problem. Chen et al.[109] employed max-norm constrained matrix completion approach for multiplex link prediction. The method proposed by Abdolhosseini-Qomi et al.[110] used layer construction using information from all layers of multiplex network for link prediction. Other solutions, like the one proposed by Zhang et al.[111], also employ GNN (graph neural networks) to use the whole graph as input rather than just the link properties, but they are not expressly optimised for multiplex networks. Nasiri et al.[112] expand the Local Random Walk (LRW) algorithm for multiplex networks by using inter- and intra-layer information to execute biassed random walks over the network, indicating the chance of certain nodes appearing. In Nasiri et al.[113], authors explore the effect of different centrality measures on link prediction performance in multiplex networks. Though we concentrate on link prediction within layers themselves in this work, interlayer link prediction is also a problem which is addressed most recently in Tang et al.[114]. In Mohapatra[115], the author proposed a link prediction approach in multiplex networks which uses both structural and spectral properties of network for link prediction. This is done by extracting both neighborhood-structural composition (using k-shell structure paradigm) and neighborhood-spectral composition (using principle eigenvectors) and average of the cosine similarity of the two.

According to Bai et al.[116], link prediction in multiplex networks is a multi-attribute decision-making issue in which possible connections in the target layer are treated as alternatives, layers as features, and the similarity score of a possible connection in each layer is a feature value. Luo et al.[31] presented a similar multiple-attribute decision-making technique, in which alternatives represent possible linkages in the target layer and characteristics represent the network's different layers. Ding et al.[117] offer a second-order iterative degree penalty (SOIDP) technique for predicting inter-layer linkages between nodes. This algorithm takes into account the information of first- and second-order common matched neighbours. Shan et al.[118] describe a machine learning-based framework in which, in addition to traditional link prediction methods that focus on particular layers for feature computation, the authors suggest two additional features that account for all levels in feature extraction. This expanded feature set is subsequently utilised to calculate edge probabilities in machine learning models. Malhotra and Goyal[119] have investigated correlations between single-layer and multiplex networks for similarity-based embedding. Quasi-local methods (L3 by Kovács et al.[27] and SHOPI by Kumar et al.[1]), which find a middle ground between complexity and accuracy, have been recommended for single-layer networks and demonstrate significantly better link prediction performance than either local or global approaches, while in this thesis an attempt is made to develop a method based on the same underlying principles for multiplex networks. Random walk-based link prediction approaches, such as those developed by Berahmand et al.[70, 71], are also classified as quasi-local link prediction methods.

2.4.2 Experimental Setup

All experiments mentioned in this work were performed on a machine with Ryzen 2700 8 core CPU, 32 GB 2666 MHz DDR4 memory, and 512 GB NVME SSD hard disk. The programming has been done on Python language version 3.6. The code for link prediction on specific layers is available on Github at https://github.com/shivansh-mishra/linkpredict-multiplex-layer. Each experiment was executed for each algorithm and testing edges to total edges percent value ($Ratio$) 100 times.

2.4.3 Datasets

For verification of the performance of the proposed link prediction approaches in this thesis, the following real world multiplex networks are used. In Table 2.2 some structural information of these datasets are listed. All multiplex networks are initially treated as unweighted and undirected. All datasets can be found in Manlio De Domenico's repository[7].

- Lazega-Law-Firm [120, 121] - This network has three layers and close to 70 nodes in all layers. These nodes represent employees of a law partnership firm and layers are different kinds of relationships between them.
- CA-Aarhus [122] - The nodes of this five-layered dataset represent workers of Computer Engineering department at Aarhus university. The five layers represent five kinds of collaborative relationships between these employees such as co-authorship etc.
- Vickers-Chan-7thGraders [123] - This three-layered dataset represents the kind of relationship between students of an Australian university.

[7] https://manliodedomenico.com/data.php

TABLE 2.2: Datasets and their basic topological characteristics

DATASET	LAYER	NODES	EDGES	AVG SHORTP.	CLUSTER COE.	ASSOR. COE.	AVG CONNECT.
Lazega-Law-Firm	1	71	717	1.76	0.52	0.02	15.42
	2	69	399	2.15	0.48	0.08	7.07
	3	71	726	1.72	0.51	-0.08	15.69
CS-Aarhus	1	60	193	3.13	0.66	0	2.49
	2	32	124	1.84	0.28	0	1.41
	3	25	21	0.83	0.11	0.02	0.03
	4	47	88	3.02	0.3	-0.01	1
	5	60	194	2.35	0.63	-0.21	3.15
Vickers-Chan-7thGraders	1	29	240	1.36	0.75	-0.16	13.33
	2	29	126	1.75	0.68	-0.15	5.47
	3	29	152	1.7	0.71	-0.11	6.83
Kapferer-Tailor-Shop	1	39	158	1.99	0.46	-0.18	5.27
	2	39	223	1.73	0.5	-0.05	8.17
	3	35	76	2.43	0.28	-0.08	2.11
	4	37	95	2.13	0.32	-0.16	2.55
CKM-Physicians-Innovation	1	215	449	3.1	0.23	-0.14	0.69
	2	231	498	3.32	0.25	-0.1	0.84
	3	228	423	3.89	0.2	0.1	0.66
Xenopus-Genetic	1	17	15	0.37	0	-0.29	0
	2	232	214	4.1	0.01	-0.13	0.02
	3	277	289	5.03	0.06	-0.19	0.09
	4	46	33	0.48	0	-0.17	0
	5	10	7	0.09	0	-0.58	0
Pierreauger	1	22	60	0.35	0.04	0.56	0.83
	2	165	550	3.56	0.28	0.59	0.84
	3	232	5433	2.49	0.41	0.64	0.93
	4	49	76	0.55	0.07	0.4	0.72
	5	71	105	0.58	0.1	0.36	0.82
	6	87	191	1.14	0.14	0.53	0.88
	7	24	61	0.33	0.04	0.66	0.9
	8	73	184	1.74	0.11	0.54	0.87
	9	45	80	0.33	0.08	0.56	0.93
	10	15	21	0.08	0.02	0.94	0.98
	11	28	51	0.72	0.04	0.36	0.74
	12	76	211	1.7	0.12	0.34	0.74
	13	26	53	0.22	0.03	0.79	0.96
	14	15	18	0.07	0.02	1	1
	15	27	38	0.15	0.04	0.79	0.92
	16	10	21	0.08	0.02	1	1
Rattus-Genetic	1	2035	2772	3.71	0.07	-0.23	0
	2	1017	982	6.8	0.01	-0.12	0.01
	3	149	116	0.42	0.01	-0.1	0.19
	4	39	21	0.03	0	-0.3	0
	5	8	5	0.01	0	-0.43	0
	6	15	9	0.01	0	-0.34	0

- Kapferer-Tailor-Shop [124] - This is one of the oldest and most researched multiplex network with four layers in which staff and customers of a tailor shop in Zambia are considered as nodes and four different interaction types are modeled as layers of the network.

- CKM-Physicians-Innovation [125] - This dataset records relationships between doctors in different towns of USA and their collaborations at the time of adoption of a new medicine.

- Xenophus-Genetic [126] - This multiplex network considers different types of genetic interactions for organisms in the Biological General Repository for Interaction Datasets (BioGRID, thebiogrid.org) for african clawed frog. The layers are of type association, direct interaction, physical association, colocalization and suppressive genetic interaction.
- Pierreauger [127] - This multiplex network is a combination of different types of tasks in the Pierre Auger Collaboration which studies cosmic rays coming from the outer space. These tasks are divided into 16 subtypes (layers) based on their keywords and content.
- Rattus-Genetic [128] - This multiplex network considers different types of genetic interactions for organisms in the Biological General Repository for Interaction Datasets (BioGRID, thebiogrid.org). The layers are of type physical association, direct interaction, colocalization, association, additive and suppressive genetic interaction.

2.4.4 Baseline Methods

For the performance evaluation and comparison of the proposed algorithms, the following baseline methods have been used. These are standard link prediction methods used on weighted networks [129] and the equations are used from that work.

1. *Common Neighbor index - weighted (CN-WT) [16]:*

$$S(n_1,n_2) = \sum_{z \in N(n_1) \cap N(n_2)} \frac{w[n_1,z] + w[z,n_2]}{2} \tag{2.21}$$

2. *Jaccard coefficient index- weighted (JC-WT) [130]:*

$$S(n_1,n_2) = \frac{\sum_{x\in N(n_1)\cap N(n_2)}(w[n_1,x]+w[x,n_2])}{\sum_{y\in N(n_1)\cup N(n_2)}(w[n_1,y]+w[y,n_2])} \tag{2.22}$$

3. *Preferential Attachment index- weighted (PA-WT) [131]:*

$$S(n_1,n_2) = \sum_{z\in N(n_1)\cup N(n_2)} \frac{w[n_1,z]+w[z,n_2]}{2} \tag{2.23}$$

4. *Adamic Adar index - weighted (AA-WT) [17]:*

$$S(n_1,n_2) = \sum_{z\in N(n_1)\cap N(n_2)} \frac{w[n_1,z]+w[z,n_2]}{\log(\sum_{x\in N(z)} w[x,z])} \tag{2.24}$$

5. *Resource Allocation index - weighted (RA-WT) [132]:*

$$S(n_1,n_2) = \sum_{z\in N(n_1)\cap N(n_2)} \frac{1}{\sum_{x\in N(z)} w[x,z]} \tag{2.25}$$

6. *Clustering coefficient index - weighted (CC-WT) [133]:*

$$S(n_1,n_2) = CC(n_1) + CC(n_2) \tag{2.26}$$

where,

$$CC(x) = \frac{1}{\triangle(x)*(\triangle(x)-1)} * \sum_{m,n\in\triangle(x)} \frac{w[x,m]+w[n,x]}{2*\sum_{z\in\triangle(x)}\frac{w[z,x]}{|\triangle(x)|}} \tag{2.27}$$

7. *Local path index - weighted (LocalP-WT) [18]:*

$$\begin{aligned} S(n_1,n_2) = &\sum_{z\in N(n_1)\cap N(n_2)} (w[n_1,z]+w[z,n_2]) \\ &+ p * \sum_{x,y\in path(n_1,x,y,n_2)} w[n_1,x]+w[x,y]+w[y,n_2] \end{aligned} \tag{2.28}$$

8. *Node Similarity Index based on Layer Relevance (NSILR-MUL)*: In Yao et al.[30], the authors proposed a method which combines current layer similarity with weighted similarity of other layers by evaluating dependence between layers using a direct matching approach.

$$S^{\alpha}(n_1,n_2) = \Big((1-\phi)*sim^{\alpha}(n_1,n_2)\Big) + \left(\phi * \sum_{\beta_i}\Big(\mu^{\alpha\beta_i} * sim^{\beta_i}(n_1,n_2)\Big)\right) \quad (2.29)$$

Here $\mu^{\alpha\beta_i}$ is layer similarity of layer β_i on layer α and $sim^{\alpha}(x,y)$ is similarity between nodes for layer α calculated using common edges (GOR). ϕ i.e., relative weightage is set as 0.5 and resource allocation is taken as similarity calculation function.

9. *Mutiple Attribute based Decision Making for multiplex networks (MADMLP-MUL)*: In Luo et al.[31], the link prediction problem is treated as a multiple attribute decision making problem with same layer similarity calculated using resource allocation index and cosine similarity is used for calculating similarity between layers.

10. *MultiVERSE - Multiplex Network Embedding (MVERSE-EMB)*: In this approach, the link prediction problem is solved using node embedding technique proposed by Pio-Lopez et al. [134], in combination with neural network based deep learning for training and predictions for edge embeddings[8].

2.5 Concluding Remarks

In this chapter the background information about the field of link prediction was presented. Also, information about the evaluation frameworks which will be followed in

[8] https://github.com/LPioL/MultiVERSE

this thesis was introduced. Further sections present the differences in link prediction on simple and multiplex networks as well as the datasets, baseline algorithms and the recent research which has been conducted in literature. All this information has helped us in identifying the research gap which in addressed in this thesis. The algorithms which have been recently proposed for link prediction in simple networks have shown the relative superiority of quasi-local link prediction methods over its more local information-based counterparts. In simple unweighted networks, the individual importance of edges has been overlooked. This importance can be extracted by evaluating the extended neighborhood of the edges themselves. Furthermore, this importance can be used at the time of local similarity-based link prediction and hence provides us with an avenue of extension towards quasi-local information-based method. This assumption has been explored in the next chapter. Edge relevance can also be exploited for link prediction in multiplex networks by summarizing the network's different layers and finding common patterns across layers and their individual applicability based on the structural overlap. This facet is also explored in further chapters of this thesis.