Extending Graph-Based LP Techniques for Enhanced Insights Into Complex Hypergraph Networks

NANDINI, YV, TANGIRALA, Jaya Lakshmi <http://orcid.org/0000-0003-0183-4093>, ENDURI, Murali Krishna, SHARMA, Hemlata <http://orcid.org/0000-0002-7566-4413> and AHMAD, Mohd Wazih

Published version

NANDINI, YV, TANGIRALA, Jaya Lakshmi, ENDURI, Murali Krishna, SHARMA, Hemlata and AHMAD, Mohd Wazih (2024). Extending Graph-Based LP Techniques for Enhanced Insights Into Complex Hypergraph Networks. IEEE Access, 12, 51208-51222.

Digital Object Identifier

Extending Graph-Based LP Techniques for Enhanced Insights into Complex Hypergraph Networks

Y.V.NANDINI[1], T. JAYA LAKSHMI[1,2](Member, IEEE), MURALI KRISHNA ENDURI[1](Member, IEEE), HEMLATA SHARMA[2], MOHD WAZIH AHMAD[3]

[1]Algorithms and Complexity Theory Lab, Department of Computer Science and Engineering, SRM University-AP, Andhra Pradesh, India.
[2]Department of Computing, Sheffield Hallam University, Sheffield, United Kingdom.
[3]Department of Computer Science and Engineering, Adama Science and Technology University, Adama, Ethiopia. (e-mail: wazih.ahmad@astu.edu.et)

Corresponding author: Mohd Wazih Ahmad (e-mail: wazih.ahmad@astu.edu.et).

ABSTRACT Many real-world problems can be modelled in the form of complex networks. Social networks such as research collaboration networks and facebook, biological neural networks such as human brains, biomedical networks such as drug-target interactions and protein-protein interactions, technological networks such as telephone networks, transportation networks and power grids are a few examples of complex networks. Any complex system with entities and interactions existing between the entities can be modelled as a graph mathematically, with nodes representing entities and edges reflecting interactions. In numerous real-world circumstances, interactions are not confined to pair of entities. Majority of these intricate systems inherently possess hypergraph structures, characterized by interactions that extend beyond pairwise connections. Existing studies often transform complex interactions at a higher level into pairwise interactions and subsequently analyze them. This conversion frequently leads to both the loss of information and the inability to reconstruct the original hypergraph from the transformed network with pairwise interactions. One of the most essential tasks that can be performed on these graphs is Link Prediction (LP), which is the task of predicting future edges(links) in a graph. LP in graphs is well investigated. This article presents a novel methodology for predicting links in hypergraphs. Unlike conventional approaches that transform hypergraphs into graphs with pairwise interactions, the proposed method directly leverages the inherent structure of hypergraphs in predicting future interaction between a pair of nodes. This is motivated by the fact that hypergraphs enable the depiction of intricate higher-order relationships through hyperlinks, enhancing their representation. Their capacity to capture complex structural patterns improves predictive capabilities. Node neighborhoods within hypergraphs offer a comprehensive framework for LP, where hyperlinks simplify interactions between nodes across cliques. We propose a novel method of Link Prediction in Hypergraphs (LPH) to predict interactions within hypergraphs, maintaining their original structure without conversion to graphs, thus preserving information integrity. The proposed approach LPH extends local similarity measures like Common Neighbors, Jaccard Coefficient, Adamic Adar, and Resource Allocation, along with a global measure, Katz index, to hypergraphs. LPH's effectiveness is assessed on six benchmark hyper-networks, employing evaluation metrics such as Area under ROC curve, Precision, and F1-score. The proposed measures of LP on hypergraphs resulted in an average enhancement of 10% in terms of Area under ROC curve compared to contemporary as well as conventional measures. Additionally, there is an average improvement of 70% in precision and around 50% in F1-score. This methodology presents a promising avenue for predicting pairwise interactions within hypergraphs while retaining their inherent structural complexity as well as information integrity.

INDEX TERMS
Link prediction; Complex hyper-networks; Hypergraphs

I. INTRODUCTION

Many real-world complex systems containing entities that interact can be modeled as complex networks [1], [2]. Graphs and hypergraphs serve as modelling frameworks for complex networks, each with its own strengths and limitations [3]. Both models contain nodes depicting real-world entities. They vary in the representation of edges. Edges in graphs represent pairwise interactions between two entities. Whereas edges in hypergraphs called as hyperedges can link multiple nodes simultaneously, allowing for interactions involving more than two entities. We use "graphs" and "networks" as well as "hypergraphs" and "hyper-networks" interchangeably in this paper. A few examples of complex networks and complex hyper-networks are given in Table.1.

Hyper-networks provide a more expressive representation for situations involving higher-order relationships among entities [4]. Hypergraphs can become computationally intensive, especially as the size of hyperedges increases, requiring careful consideration for scalability. The simplicity of graph representation typically leads to computationally efficient and scalable algorithms for certain types of analytical tasks. Therefore, graphs are more popular representation of complex systems. It is a common practice to transform the hypergraphs into graphs to perform any task on them. For instance, consider a network representing co-author relations between authors. Nodes of such graphs denote authors and an edge forms between nodes exist if authors co-author a research article. Consider the scenario where Paper 1 is authored by authors A, B and C together, Paper 2 is authored by authors C and D and Paper 3 is authored by authors D and E. This information can be modeled naturally as hypergraph as shown in Fig.1. The transformed network with pairwise interactions is shown in Fig.2. The collaboration network shown in Fig.2 only depicts the collaboration between pairs of authors, losing the collaboration information of a group of authors on a single publication. Thus, a collaboration situation described above can be more meaningfully represented as a hypergraph rather than graph. This clearly demonstrates that hypergraphs offer a more meaningful way to model scenarios that involve higher-order relationships among entities compared to graphs.

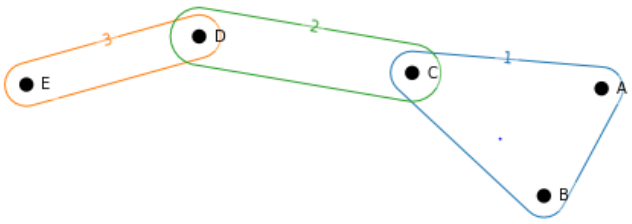

FIGURE 1: Coauthorship hyper-network denoting interactions among group of authors

Link prediction (LP) is one of a fundamental problem focusing on the estimation of the probability of a future

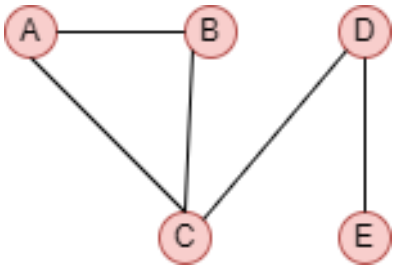

FIGURE 2: Coauthorship network representing pair-wise interaction between authors

interaction between two entities [5]. Some of the potential applications of LP are given below.

- Collaborative networks: LP algorithms, are used to predict future collaborations between authors, as well as to recommend collaboration between authors.
- Drug-target interactions: Studies on the impacts of possible drug interactions require large samples, extensive time and high cost. The interactions between drugs to predict poly-pharmacy interaction can be predicted in complex drug-target hyper-networks and most probable ones can be experimented to save cost and time.
- Transportation domain: Applications of LP in transportation domain involves:
  - Route Planning: Predict future connections to optimize route planning for vehicles.
  - Infrastructure Planning: Anticipate future connections to inform infrastructure planning and development, such as the construction of new roads or transportation hubs.
  - Traffic Management: Predict links to improve traffic management strategies, including congestion mitigation and adaptive traffic signal control.
  - Emergency Response: Predict links to improve emergency response strategies, such as rerouting traffic during accidents or natural disasters.
- Biological Networks: LP methods are utilized for predicting interaction between proteins in a protein-protein interaction networks.
- Social Networks: Facebook uses LP algorithms to recommend friends to users.
- E-Commerce: LP algorithms can be used by E-commerce websites to recommend products to users.

LP in graphs is well explored in the literature. Thus far, hypergraphs have been transformed into graphs and LP measures have been employed on the transformed graph. Converting hypergraphs to graphs provides a practical means for leveraging existing graph-based tools and algorithms. However, this may lead to significant loss of information. LP using hypergraphs offers a number of advantages. Hypergraphs enable the depiction of intricate higher-order connections through hyperlinks, making them more significant models. Their capacity to comprehend and analyse complex structural patterns enhances their forecasting abilities. The studies

TABLE 1: Various complex networks

Network	Nodes	Edges	Hyperedges
Social Networks	Individuals	Social connections, friendship etc, between two individuals	Group interactions among individuals
World Wide Web	Web pages	Hyperlinks between web pages	Group of web pages having similar content
Biological Networks	Molecular entities(genes, proteins, metabolites)	Interaction between two proteins, Regulatory relationships between two genes, Metabolic reactions	Multi-way relationships present among genes and proteins etc.
Transportation Networks	Locations	Road/transportation connecting two locations	Transportation routes connecting many locations
Internet	Routers, servers, or individual devices	Physical or logical connections between network elements	Hyperedges connecting multiple nodes
Citation Networks	Articles	Citations between two papers	Group of articles citing a single source
Collaboration Networks	Authors	Co-authorship relationships between two authors	Group of authors of an article
Epidemiological Networks	Individuals	Contacts between individuals	Interaction among group of individuals
Food-web	Species	Predator and Prey	Group of species that compete for common prey
E-commerce	Users	Transactions between two users	Coordinated actions of more than two users, such as a buyer, seller and broker

conducted by [6] and [7] demonstrated the effectiveness of employing graph cliques for LP in a graph. However, the process of finding cliques in graphs is computationally complex. The hyperlinks in hypergraphs are analogous to cliques in graphs. Therefore, Hypergraphs provide a more efficient and powerful platform for LP due to their inherent structural features. Hence, this study focuses on investigating the efficacy of directly predicting links from hypergraphs, without the need to convert them into graphs. . This motivation leads to the following research questions:

1) Could LP in hypergraphs directly yield more advantages than converting the current hypergraph model into a graph format for predicting pair-wise links?
2) What are the modifications to existing LP measures to make them adaptable to hypergraphs?

The following are the contributions made in this work:

1) A novel methodology termed Link Prediction in Hypergraphs (LPH) is proposed to predict pairwise interactions in hypergraphs, without transforming hypergraphs into graphs. This preserves the original hypergraph structure without information loss.
2) The local similarity measures of Common Neighbors, Jaccard Coefficient, Adamic Adar, Resource Allocation and a global similarity measure of Katz index are extended to hypergraphs.
3) The proposed approach of LPH is evaluated on six benchmark hyper-networks using Area under ROC curve, Precision, and F1-score.

Table. 2 describes the notation used in this work.

The structure of this document is as follows. Section II gives mathematical definitions pertaining to networks, hyper-networks, and the LP problem. Section III specifically examines the literary works related to the topic of LP in networks and hyper-networks. Section IV provides a comprehensive explanation of the proposed approach, while The results are analyzed and discussed in detail in Section V. Section VI concludes the work. Section VII derives the abbreviations used in this paper.

TABLE 2: Notations used in research work

Notation	Description
G	Graph
H	Hypergraph
V	Node set
E	Edge/Hyperedge set
p,q,r	Nodes in network
N(p)	Neighbors of node p
A	Adjacency-Matrix of G
I	Incidence-Matrix of H
k(r)	Degree of node r
D_v	Diagonal element
W	Diagonal hyperedge size
n	Number of n nodes
m	Number of m hyperedges
$\delta(s)$	Degree of hyperedge s
c_s	Cardinality of the hyperedge s

II. PROBLEM DEFINITION

This section provides the mathematical notations utilized in this work.

A. DEFINITIONS

Definition 1: **Complex Network:** Graph $G = (V, E)$ is used to describe a complex network. $V = \{v_1, v_2, .., v_n\}$ represents a collection of n nodes, and $E = \{e_1, e_2, .., e_m\}$ represents a set of m edges.

Definition 2: **Complex Hyper-network:** A complex hyper-network is represented as a hyper-network $H = (V, E)$. $V = \{v_1, v_2, .., v_n\}$ collection of n nodes, and $E = \{E_1, E_2, .., E_m\}$ collection of m hyperedges, where each hyperedge $E_i \in (2^V - \phi)$.

TABLE 3: Edge function in graph and hyper-networks

Types of graph	Vertex set	Edge set
Graph	$V = \{v_1, v_2, \ldots, v_n\}$	$E \subseteq VXV$
Hyper-network	$V = \{v_1, v_2, \ldots, v_n\}$	$E \in 2^V$

Both these networks can be represented using matrices. Adjacency-Matrix is the well known matrix representation

for a graph, as the interactions are pair-wise, where as hyper-network is denoted by incidence matrix because of interactions among group of nodes. These two matrices are given below.

Definition 3: **Adjacency-Matrix**: The Adjacency-Matrix $A(G)$, of a graph $G = (V, E)$ is square matrix where entries a_{pq} indicate edge counts involving nodes v_p and v_q. The diagonal elements of $A(G)$ are consistently zero. This matrix construction can be achieved by:

$$A_{pq} = \begin{cases} 1 & : if(v_p, v_q) \in E \\ 0 & : otherwise \end{cases} \quad (1)$$

The adjacency matrix of Fig.2 is given below:

$$\begin{array}{c} \\ \text{A} \\ \text{B} \\ \text{C} \\ \text{D} \\ \text{E} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{A} & \text{B} & \text{C} & \text{D} & \text{E} \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

Definition 4: **Incidence-Matrix**: In incidence matrix I, is of size $m \times n$. The vertices are represented by the n rows, and the hyperedges by the m columns. I can be constructed as follows:

$$I_{ps} = \begin{cases} 1 & : if\ node\ p\ is\ part\ of\ hyperedge\ s \\ 0 & : otherwise \end{cases} \quad (2)$$

The incidence matrix of Fig.1 is given below:

$$\begin{array}{c} \\ \text{A} \\ \text{B} \\ \text{C} \\ \text{D} \\ \text{E} \end{array} \begin{array}{c} \begin{array}{ccc} \text{e1} & \text{e2} & \text{e3} \end{array} \\ \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \end{array}$$

Note that, a graph is a specific type of hypergraph where each of its edges has a cardinality of 2. Adjacency-Matrix can be built from the corresponding incidence matrix using the equation.

$$A = IWI^T - D_v \quad (3)$$

The diagonal elements of the matrix D_v, which represents the nodes' degrees, I^T, the transpose of the incidence matrix and W is the diagonal hyperedges size.

B. LINK PREDICTION PROBLEM

Complex hyper-networks evolve as nodes, and edges (links) are added/removed over time. Hence, forecast future-links or identify any missing-links in a network is vital for its evolution. LP in hyper-networks is the problem of predicting future hyperlinks.

Definition 5: **Link Prediction in Hyper-networks (LPH):** Given a hyper-network $H = (V, E)$, V representing set of vertices, and E denoting set of hyperlinks, the problem of LPH is to predict hyperlinks which are not existing in H, but predicted to appear in future. Fig.3 illustrates the problem where the size of hyperlinks is restricted to be 2.

Commonly used technique is to transform a hypergraph to a graph using Eq.3, then make predictions about future interactions in the transformed graph. This prediction is restricted to the prediction of interactions between pair of nodes. Notably, our approach LPH aims to forecast the emergence of the hyperedges $e2$ and $e3$, delineated by dashed lines, directly without transforming the hypergraph into graph.

Consider Fig.3. There are 7 nodes denoted by $A, B, C, D, E, F,$ and G, and 4 hyperedges labeled as $e0, e1, e4,$ and $e5$. Hyperedges $e0$ and $e1$ involve more than two nodes, specifically including E, F, G and $C, D, E, F,$ nodes respectively. Hyperedges $e4$ and $e5$ consist of pairwise nodes, namely A, B and B, E.

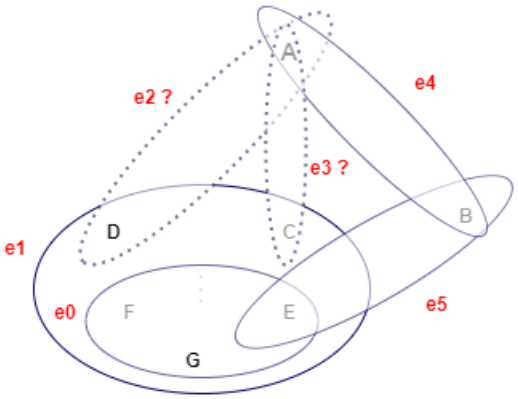

FIGURE 3: An illustration for Link Prediction in Hypergraphs

The objective of the LPH problem is to predict hyperlinks that are currently absent but anticipated to emerge in the future. In Fig. 3, we aim to know whether the nodes A and D as well as A and C interacts in

Definition 6: **LP (LP):** The LP issue for graphs was defined by Liben-Nowell et al. [8] in the following way: Given a Graph, $G(V, E)$, with set of nodes V and set of edges E representing network during time interval t, LP requires creating a list of edges not present in $G[t_0]$, but is expected to appear in the network $G[t_1]$ where $t_0 < t_1$.

Fig.4 illustrates the problem for LP in graphs. The network at time t_0 includes 5 nodes and 7 edges. By time t_1, two edges have been introduced between nodes A, B and C, D. LP forecasts the netwrok's future edges, specifically at time t_n.

The problem of LP is challenging in both cases because of enormous magnitude of the candidate node pairs between which the interaction is to be predicted. When predicting links in hypergraphs, the size of the candidate set for LPH is $2^V - E$. A few researchers restrict the hyperlink size to be upto some number k and limit the problem as $k - regular$ [9]. In this work, we fix k to be 2. While considering LP in graphs, the possible edges are $|V|(|V| - 1)$ where as

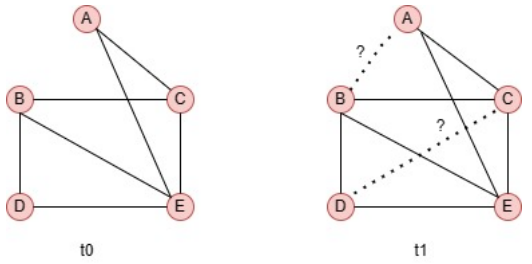

FIGURE 4: An example for LP problem in graphs

the existing edges (E) are very few. Therefore, the candidate node pairs for predicting possibility of future link is $|V|(|V|-1)-|E|$, which is very large in number. There are several advantages of predicting pair-wise links in hypergraphs rather than in the transformed graph. Hence, this study proposes using hypergraphs directly for the task of LP, bypassing the need for their transformation into conventional graphs.

III. LITERATURE

Nodes are entities within a complex network that possess characteristics such as keywords, research interests, and demographic information in coauthorship networks, personal and professional information in Facebook and LinkedIn networks, respectively. The properties of a nodes are denoted as a vector, and the similarity between the nodes can be calculated using a distance measure like Euclidean or cosine. Such measures present two challenges. The first is that the qualities vary depending on the domain, other is privacy. Domain-dependent measurements are not applicable to all types of networks, privacy concerns sometimes prevent node properties from being made public. As a result, other measurement is based on structural similarity of nodes within a graph gained popularity. LP in graphs is well explored in the literature. Section III-A briefly explains the popular measures for LP in graphs. However, there are limited works on LP in Hypergraphs, which are specified in Section III-B. The existing literary works for LP in graphs and hypergraphs are summarised in Fig.5.

A. LP IN GRAPHS

LP measures use heuristics to compute and award a particular score to a non-adjacent node pairs. The nodes with the highest score are the most likely to join together in the future. Based on the computation of LP score, these measures are classified into similarity-based measures, probabilistic-based measures, dimensionality-based measures and other measures.

1) Similarity-based measures

These metrics use the graph's structural characteristics to calculate the score between two nodes. Those methods which depend on immediate neighborhood of the nodes are called as local similarity measures. There is another class of similarity measures, which consider entire graph topology to compute score. These are called as global similarity measures. Quasi local measures exploit these two measures by exploiting the strengths of both [10].

- **Local similarity measures:** Common-Neighbors (CN), Jaccard-Coefficient (JC) , Adamic-Adar (AA), Resource-Allocation (RA), Preferential-Attachment (PA), are a few popular measures in this category.
  Common-Neighbors : When two nodes share a significant number of common neighbors, the possibility of forming a link increases [11]. The equation for common neighbors between two non-adjacent nodes p and q abbreviated as $CN_{p,q}$ is as follows:

$$CN_{p,q} = |N(p) \cap N(q)| \tag{4}$$

where the set of nodes p and q's neighbors is represented by $N(p)$ and $N(q)$, respectively.
Jaccard-Coefficient: The Jaccard-Coefficient is the normalized Common-Neighbor. The Jaccard-Coefficient is calculated by dividing the total number of different neighbors that either node has, by the number of neighbors that both nodes share [12].

$$JC_{p,q} = \frac{|N(p) \cap N(q)|}{|N(p) \cup N(q)|} \tag{5}$$

where $N(p), N(q)$ are neighborhood sets of nodes p,q. The Jaccard Coefficient is favored over Common Neighbors when considering differences in node degrees and ensures that the similarity measure is not biased towards nodes with higher degrees. Jaccard Coefficient's robustness to change in network size and density compared to Common Neighbors, handles sparsity effectively. The Jaccard Coefficient's normalized measure ranging between 0 and 1, representing the overlap between nodes' neighborhoods intuitively.
Adamic-Adar : By giving the less-connected neighbor a higher weight, this index outlines how the basic counting of common neighbors might be improved, and is defined as [13]:

$$AA_{p,q} = \sum_{r \in N(p) \cap N(q)} \frac{1}{log|k(r)|} \tag{6}$$

where $k(r)$ is the degree of node r.
Resource-Allocation : In order to evaluate their similarity, let us assume that node p delivers resources to q equally across their common nodes [14]. The similarity between nodes rises with the volume of resources transmitted between them. Mathematically, can be represented as :

$$RA_{p,q} = \sum_{r \in N(p) \cap N(q)} \frac{1}{k(r)} \tag{7}$$

the degree of node r is denoted by $k(r)$.

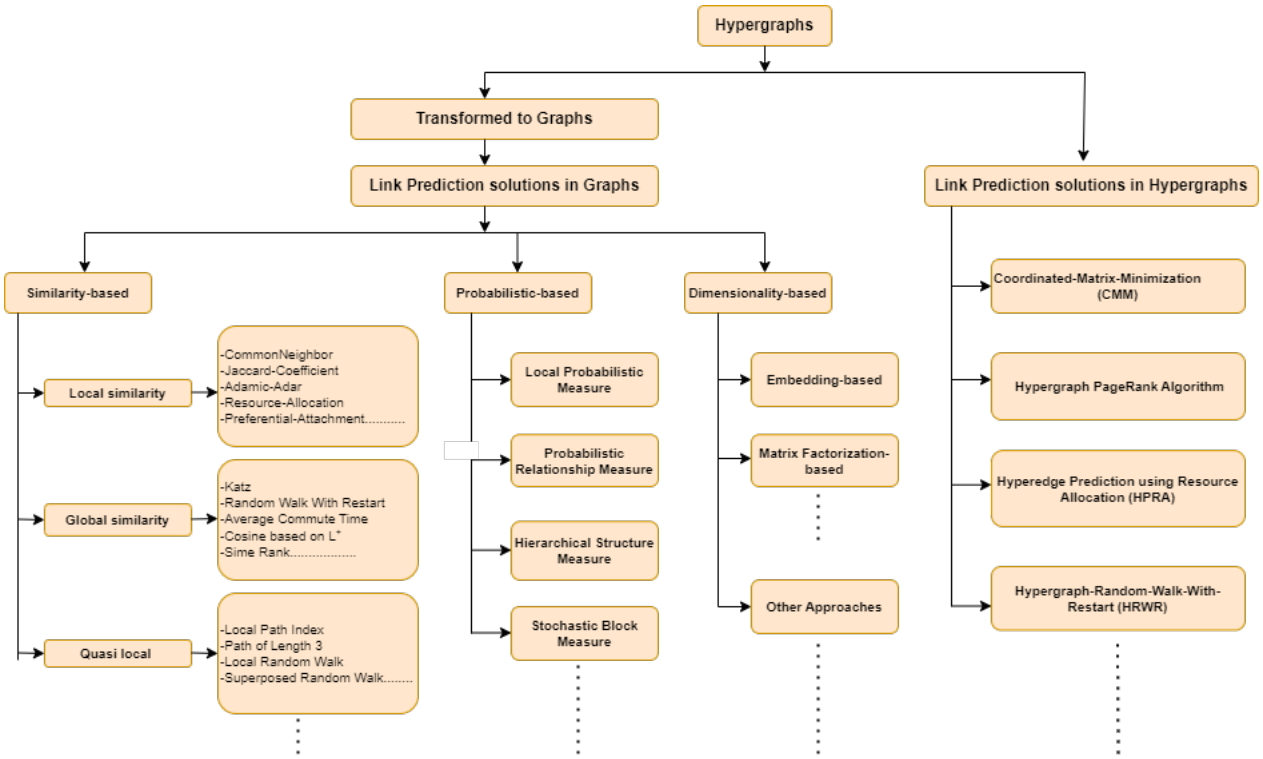

FIGURE 5: Exploring Predictive Models for Graphs and Hypergraphs: A Literature for Link Prediction

Preferential-Attachment : The degrees of nodes p and q together can be multiplied, which finds the richness of two nodes [15].

$$PA_{p,q} = |N(p)| * |N(q)| \tag{8}$$

where the set of nodes p and q's neighbors is represented by N (p) and N (q), respectively. PA needs the degree of nodes and does not consider common neighbors.
Beyond the ones listed, the literature contains plenty of additional local similarity measurements.

- **Global similarity measures**: Global similarity measures are mainly focused on shortest-paths and random walks in the graph. As the paths and walks can easily computed based on the Adjacency-Matrix (A) of a graph, many of the global measures use A in their computation. Katz-Index (KZ) given below is the most popular measure in this category, which is computed based on paths in the graph.
Katz-Index : Katz-Index aggregates weighted sum across all shortest paths between p and q by penalizing longer paths with a damping factor $0 < \beta < 1$ [16]. The equation is derived as:

$$KZ_{p,q} = \sum_{l=1}^{\infty} \beta^l |paths_{p,q}^{<l>}| = \sum_{l=1}^{\infty} \beta^l (A^l)_{p,q} \tag{9}$$

where $paths_{p,q}^{<l>}$, set of total l length paths between p and q, β the damping factor which gives more weights to shortest paths, A is a graph's Adjacency-Matrix, $\beta < \frac{1}{\lambda}$, wherein λ represents the highest eigenvalue of matrix A.
Random Walk With Restart (RWR) [17], Average Commute Time (ACT) [18] and Sim Rank [19] are the popular random walk based measures. All these use variations of number of steps taken to reach from one node to the other in a random walk.
Local and global measures have their advantages and limitations. There is another category of quasi local, which use the advantages of both the categories. Local Random Walk(LRW), LRW assesses the similarity between node pairs by focusing on limited step random-walks [18]. Superposed Random Walk (SRW) is another quasi local measure that assigns highest score to the nearest nodes. While using a local random walk [18]. Local Path Measure(LPM) is path-based quasi local measure that computes the similarity scores between node pairs using paths of length two and three [12].

2) Probabilistic-based measures

These measures compute the scores between a pair of nodes, depending on the statistical probabilities of the nodes. More information is typically needed for probabilistic models, such as node or edge attributes, along with structural information to compute the statistics. Local Probabilistic Measure (LPM) produces three types of features, and are derived from several information sources: topological, semantic, and co-occurrence probability characteristics. Probabilistic Relational Measure (PRM) provides both node and link attributes [20]. The next measure, Probabilistic Entity Relationship Measure (PERM) uses directed arcs to define relation be-

tween attributes. Hierarchical Structure Measure (HSM), where most of the networks are hierarchically structured, in which nodes split into groups and subgroups, and the subgroup information is used to predict links. The next one is Stochastic Block Measure (SBM), it uses tensor interactions to provide a stochastic framework for entity connections.

3) Dimensionality-based measures

These metrics calculate the score between two nodes using a function F, that operates directly on an Adjacency-Matrix or a graph's Laplacian matrix [20]. Popularly using dimensionality measures are Embedding-based, Matrix Factorization-based measures.

Embedding-based measure: It is a dimensionality reduction measure, it maps higher D dimensional nodes to lower d dimensional nodes in the graph by preserving node neighborhood structure.

$$Emb_{p,q} \approx Z_p^T Z_q \tag{10}$$

where the d-dimensional embedding of the node p is denoted by Z_p, and embedding-matrix, $Z \in R^{d \times |V|}$, where each column representing embedding vector of particular node, $Emb_{p,q}$ is a function which computes pairwise similarity scores generated from embedding.

Matrix Factorization-based measures: Matrix factorization is used in lots of LP papers since last decade [21]. Mostly, researchers extracted latent-features and using these feature nodes in supervised/unsupervised LP. Adding more nodes, links, or attribute data can help the prediction results even more. Some authors used both, non-negative-matrix-factorization, singular-value-decomposition [22]. The matrix $X = (x_1, x_2.., x_n)$, that has columns with n data vectors. Now generalizing the matrix to:

$$X \approx FG^T \tag{11}$$

where $X \in R^{p \times n}$, $F \in R^{p \times k}$. Hence F called as basis matrix, and coefficient matrix is denoted by G, whereas k represents the dimension of latent space $(k < n)$. Few popular matrix factorization techniques are listed here, Singular-Value-Decomposition (SVD) [23], Non-Negative-Matrix-Factorization (NMF) [24], Semi-NMF [25].

There are so many other approaches for LP such as machine learning-based measures, Clustering-based measures and Information Theory-based measures, which can be found in [21].

B. LINK PREDICTION IN HYPERGRAPHS

We review hyperlink prediction techniques from [26]. These techniques are classified into similarity-based [27], probability-based [28], matrix optimization-based methods [29] for hyperlink prediction.

1) Similarity-based methods in hyperlink prediction

In Similarity-based methods in hyperlink prediction, rather than pairwise connections between nodes as in graphs, hyperedges are used to represent relationships involving more than two nodes in hypergraphs. These measures compute the score among the nodes based on structural attributes of hyperedges and also considers the nodes they connect. Some few popular measures are Common-Neighbors, Katz, Hyperlink Prediction using Resource Allocation.

- **Common-Neighbors in hyperlink prediction:** A node's degree in a hypergraph indicates how many hyperedges are connected to it, and the number of neighbors is the total number of nodes that a particular node shares at least one hyperedge with. The CN in hyperlink prediction is defined as:

$$CN_r = \frac{2}{c_s\left(c_s - 1\right)} \sum_{p,q \in s} CN_{p,q} \tag{12}$$

  where c_s is the cardinality of the hyperedge s.
  Similar to CN, KI may be extended to hyperlinks by substituting the hypergraph adjacency matrix for the graph adjacency matrix A. $A = IWI^T - D_v$ is a common definition for a hypergraph's adjacency matrix.
- **Hyperlink Prediction using Resource Allocation:** Based on the ideas of the resource allocation process, hyperlink prediction using resource allocation, or HPRA, is a newly developed direct hyperlink prediction technique. HPRA uses the direct link and common neighbors between two nodes to calculate the hypergraph resource allocation (HRA) index. HRA between two nodes:

$$HRA_{p,q} = SC_{p,q} + \sum_{r \in N(p) \cap N(q)} \frac{SC_{pr} \times SC_{rq}}{k(r)} \tag{13}$$

  where $SC_{p,q} = \sum_{s \ni p,q} \frac{1}{c_s - 1}$, $k(r)$ is the degree of node r, the set of nodes p and q's neighbors is represented by $N(p)$ and $N(q)$, respectively.

2) Probability-based methods in hyperlink prediction

These measures consider the structural characteristics of the hypergraph, much like graphs do. With hypergraphs, on the other hand, the analysis goes beyond pairwise connections to take relationships represented by hyperedges into account. Three probability-based techniques for hyperlink prediction are examined by the author.

- **Node2Vec:** Node2Vec follows random-walk technique which investigates neighborhoods using both depth-first and breadth-first sampling strategies [30]. Node2Vec for hyperlink prediction is computed as [9]:

$$S_{N2V} = sigmoid\left(\frac{1}{c_s} \sum_{v_p, v_q \in s, p \neq q} x_i^T x_j\right) \tag{14}$$

  that produces probabilistic measures that counts the average correlation between pair of nodes in hyperlink s. The final existence confidence of hyperlink s is indicated by score S_{N2V}.
- **Bayesian Set:** Using probability, the Bayesian Set (BS) method retrieves objects from a cluster. While retrieving, only few items are retrieved from cluster and the

problem is handled as a Bayesian inference issue [31]. This approach uses a model based understanding of clusters to provide a score to each item based on, how likely it is, that the item will be found in a cluster which includes the items with queries. Let D represent a set of data elements, and let D_c be a collection of queries such that $D_c \subset D$. Once D_c has been seen, the items score $p \in D$ that belongs to D_c, that can be defined as:

$$S_{BS} = \frac{pb(p, D_c)}{pb(p), pb(D_c)} \tag{15}$$

In numerator, the probability for p, D_c is generated from same method with same attributes, while denominator is the probability that p, D_c generated from same method with different attributes, where D and D_c are known hyperlink sets.

- **Hyperlink Prediction Using Latent Social Features (HPLSF):** The first machine learning technique created for hyperlink prediction is HPLSF [32]. When producing latent node characteristics, HPLSF eliminates all higher-order topological attributes and solely takes into account the pairwise distances between nodes.

3) Matrix optimization-based methods in hyperlink prediction

Numerous matrix optimization-based hyperlink prediction techniques are being studied by researchers. Spectral Hypergraph Clustering (SHC), Matrix Boost (MB), and Coordinated Matrix Minimization (CMM) are a few popular techniques in this. The incidence, adjacency, or Laplacian matrices/tensors of hypergraphs are essentially used in these techniques to frame matrix optimization problems for hyperlink prediction.

- **Spectral Hypergraph Clustering:** SHC aims to learn a partition in which, the links among several nodes within the same group are dense, whereas the links between two groups are sparse [29]. Given a Hypergraph H, where n defines nodes and the SHC model is defined as:

$$\min_f ||f - y||_F^2 + \mu f^T L f \tag{16}$$

let $f \in R^n$ is characterized as function of classification, $y \in R^n$ is the vector with label that contains values of 0, 0.5, 1, $\mu > 0$ is the parameter which regularizes the function and the hypergraph's normalized Laplacian matrix is denoted by L.

$$L = Z - D^{-\frac{1}{2}} H D_v C^{-1} H^T D^{-\frac{1}{2}} \in R^{n \times n} \tag{17}$$

where $Z \in R^{n \times n}$ is the identity-matrix, $D_v \in R^{n \times n}$ is the diagonal matrix of hyperlink weights.

- **Matrix Boost:** Matrix Boost (MB) uses an iterative completion-matching optimization, to execute inference concurrently in the incidence and adjacency spaces [33]. Considering an incomplete n-node hypergraph H, denoted by $A = HH^T \in R^{n \times n}$, as the adjacency-matrix of H.
- **Coordinated Matrix Minimization:** As an alternative, CMM uses least square matching and non-negative matrix factorization in the adjacency space to identify which subset of candidate hyperlinks might be most suited to fill the needed hypergraph [27]. Like MB, indicate $A = HH^T \in \mathbb{R}^{n \times n}$, $U \in \mathbb{R}^{n \times \tilde{m}}$ as an adjacency-matrix of H, candidate hyperlinks incidence-matrix, correspondingly. Let $Q \in \mathbb{R}^{n \times k}$ a non-negative matrix, the latent factor matrix, and assumed that the adjacency-matrix of the complete hypergraph is factorized by

$$A + U \Lambda U^T \approx Q Q^T \tag{18}$$

where $\Lambda \in \mathbb{R}^{\tilde{m} \times \tilde{m}}$ is a potential hyperlink candidate diagonal indicator matrix.

Many existing works in the hypergraph literature focus on predicting hyperlinks rather than pair-wise links. However, there are very limited works focus on pair-wise LP in hypergraphs. Kumar et al. [28] proposed HPRA extends the LP measure of resource allocation to hyperlink prediction without generating candidate hyperlinks set. HPRA is a local-similarity-based measure based on the principle of the resource allocation. Along with recovering missing hyperedges, author demonstrates that HPRA predicts future hyperedges in a wide range of hypergraphs. On experimentation, HPRA gives best performance compared to existing ones. Wang et al. define random walk notion on hypergraphs and use it for hyperlink prediction [34]. Chitra et al. extended PageRank algorithm for hyperlink prediction and applied it to disease-gene hyper-network [35]. Zhang et al. introduced the Coordinated-Matrix-Minimization (CMM) algorithm [27]. This algorithm utilizes alternating non-negative-matrix-factorization and least-square-matching to deduce potential-hyperlinks within the node adjacency-space of the hyper-network. Wang et al. use drug combination data and created a model called Hypergraph-Random-Walk-With-Restart model to predict effective drug combinations [34]. A survey on hyper LP can be found at [26]. Nasiri et al. presents the Multiplex Local Random Walk (MLRW), an extension of the local random walk for LP in multiplex networks. Author utilized information from inter-layer and intra-layer interactions to develop a biased random walk [36]. The work such as [36] discusses LP in multi-relational networks, but in the hypergraph scenario, multi-relational networks become so tedious. Berahmand et al. devised a comprehensive deep semi-supervised community detection (DSSC) approach for complex networks. This method incorporates a semi-autoencoder (SEAE) along with a specified pair-wise constraint matrix derived from point-wise mutual information (PMI) within the representation layer [37]. LP based on Community structure of the network provide more prediction quality in some cases. Work of [37] discuss this scenario. However, community detection algorithms in hypergraph are not much available in the literature, and our focus is on link predictions for hypergraphs. Shang et al. examine the consensus dynamics across temporal hypergraphs, which include non-linear modulating

functions, topology varying over time, and random perturbations [38]. This work discusses temporal hypergraphs, but our focus is on paiwise interactions in hypergraphs. Shang et al. [39] investigates a three-body consensus model incorporating higher-order network interactions and social homophily principles which focus on neighbors decissions. Shang et al. [40] examines consensus formation in directed hypergraphs, extending standard graph structures to incorporate neighbor-dependent synergy in social dynamics, by using petri net method. In this work, our focus is only on undirected and unweighted hypergraphs. Exploring directed and weighted versions will be our future work.

IV. PROPOSED APPROACH

A. MOTIVATION

LPH has the following advantages:

1) Due to the provision of modeling the presence of intricate higher-order relationships in the form of hyperlinks, hypergraphs are more meaningful models.
2) The inclusion of higher-order relationships and the ability to capture richer structural patterns can contribute to enhanced predictive power in hypergraphs.
3) The concept of node neighborhood in hypergraphs provides a comprehensive framework for the task of predicting the link. For example, a hyperlink in a hypergraph can be viewed as a clique in a graph. It is straightforward to predict the interaction between two nodes, each belonging to separate cliques with common nodes in two cliques, in a hypergraph. The works such as [6] [7] are greatly benefited by this cliques in the prediction tasks. In graphs, identification of clique itself is NP-complete.

Hence, instead of transforming hypergraphs into graphs, this study proposes predicting pair-wise links directly from hypergraphs. The broad approach is depicted in Fig.6. Two approaches were employed for predicting links in hypergraphs. Initially, the hypergraph was divided into train and test sets, and the train set was converted into a pairwise graph. LP measures were then applied and evaluated against the hypergraph test set. Alternatively, the task focused on directly predicting size=2 hyperlinks from the hypergraph train set, followed by evaluation against the test set.

We choose measures such as CN, JC, AA, RA, and KZ to be extended to hyperlink prediction scenario because these serve as straightforward methods for computing scores between non-existing links, without considering any attribute information. We aim to utilize these widely recognized similarity measures as a starting point for our research, with plans to expand our investigation to include additional measures in the future.

B. LINK PREDICTION IN HYPERGRAPHS

The hypergraph is initially partitioned into a train graph and a test graph. We propose to extend the LP measures of Common-Neighbors (CN), Jaccard-Coefficient (JC),

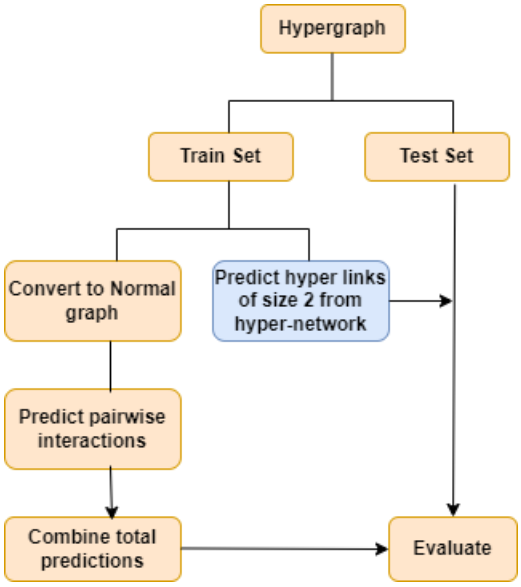

FIGURE 6: Novel Approach: Direct Prediction of Pairwise Links from Hypergraphs without Graph Transformation

Adamic-Adar (AA), Resource-Allocation (RA) and Katz Index (KZ) to predict links directly from the hypergraph. We name these extended measures as LPHRA, LPHCN, LPHJC, LPHAA, and LPHKZ. Algorithm.1 gives a concise overview of the comprehensive methodology for the proposed LPH.

Similarity-based methods for LP in graphs (Fig.5) calculate similarity scores based on graph topology for each non-adjacent node pair p and q, scores are sorted, and it assumed that the pairing with the highest scores would form future links. Few popular similarity-scores are common-neighbors, jaccard-coefficient, adamic-adar, resource-allocation, and so on. We extend these graphs similarity based LP measures to hypergraphs as follows.

1) Link Prediction in Hypergraphs using Common Neighbors (**LPHCN**)

We extend the notion of common neighbors (CN) given in Eq.4 to hypergraphs through computing the average of the pairwise CN indices between the nodes within each hyperlink [26]. The $LPHCN$ is defined as given the Eq.19.

$$LPHCN_{p,q} = \frac{2}{c_{s_1} * c_{s_2}} \sum_{p \in s_1; q \in s_2; r \in s_1 \cap s_2} |r| \qquad (19)$$

where, s_1 and s_2 are hyperedges and c_s is size of s.

2) Link Prediction in Hypergraphs using Jaccard Coefficient (**LPHJC**)

$LPHJC$ is the normalized $LPHCN$. The LPHJC is calculated by dividing the total number of different neighbors that either nodes has, by the number of neighbors that both nodes share.

$$LPHJC_{p,q} = \frac{LPHCN_{p,q}}{|N(p) \cup N(q)|} \qquad (20)$$

Algorithm 1 Link Prediction in Hyper-networks (LPH)

Input:

- $Hypergraph$
- $Trainset$: A part of $Hypergraph$ used to compute the LPH measures
- $Testset$: Remaining part of $Hypergraph$ used for performance evaluation of LPH measures.

Output: $E^{'}$: List of node pairs with probable future links.

Get $Hyperlink$ Degree Distribution
Choose size of candidate hyperlink probabilistically based on hyperlink degree distribution. Let the size chosen be k.
HL = empty list of size k // Initialize hyperlink
q = node with highest degree
HL.append(q)
for $i = 2$ to $k - 1$ **do**
  // Add k-1 nodes to hyperlink
  max_lp_score=0
  // compute node q with highest LP score with the nodes in HL using the procedures specified in section IV-B3 as follows
  for p in HL **do**
    for r in $(V - HL)$ **do**
      $lps = LP\_score(p, r)$
      if $lps > max\_lp\_score$ **then**
        $q = r$
      end if
    end for
  end for
  HL.append(q)
end for
for t in $Testset$ **do**
  $X = HL \cap t$
  if $|X| \geq 2$ **then**
    Add the 2-size subsets from X to $E^{'}$
  end if
end for
return $E^{'}$

where $LPHCN_{p,q}$ is taken from Eq.19, where the set of nodes p and q's neighbors is represented by $N(p)$ and $N(q)$, which intersects two nodes, divided by total number of common neighbors in between two nodes.

3) Link Prediction in Hypergraphs using Resource Allocation (LPHRA)

$LPHRA$ predicts pair-wise links using the principles of resource allocation. This method is inspired by the work of [28]. Contrary to graphs, hypergraphs allow nodes p and q to already be part of another hyperlink. Consequently, a resource at node p can be transmitted to node q either directly or via common neighbors. Therefore, the amount of resource transferring between node p and q is generated by:

$$LPHRA_{p,q} = \sum_{p \neq q} \frac{1}{c_s - 1}, \text{if } p, q \in s$$
$$= \sum_{r \in N(p) \cap N(q)} \frac{1}{k(r)} * \frac{1}{c_{s_1} - 1} * \frac{1}{c_{s_2} - 1}, \text{otherwise} \quad (21)$$

where s,s_1 and s_2 are hyperlinks; c_s,c_{s_1} and c_{s_2} are their sizes; r is a common neighbor of p and q; $k(r)$ represents the node r's degree and $p, r \in s_1$ and $r, q \in s_2$, the set of nodes p and q's neighbors are represented by $N(p)$ and $N(q)$.

The first component of Eq.21, is the amount transferred between p and q if both these nodes are a part of a hyperlink s. In case they are not part of same hyperlink, this component calculates to zero. Second part computes the amount of resource transmitted via all common neighbors between the two nodes.

4) Link Prediction in Hypergraphs using Adamic Adar (LPHAA)

In hypergraphs, Adamic-Adar measures how similar two nodes are to each other by looking at shared hyperedges and the common neighbors' inverse logarithmic degree centrality.

$$LPHAA_{p,q} = \sum_{p \in s_1; q \in s_2; r \in s_1 \cap s_2} \frac{1}{log(|k(r)|)} \quad (22)$$

where $|k(r)|$ represents the node r's degree in the hypergraph.

5) Link Prediction in Hypergraphs using Katz Index (LPHKZ)

Katz Index can be extended to hypergraphs by substituting the graph's Adjacency-Matrix A, with the hypergraph's Adjacency-Matrix. The computation formula is shown in Eq. 23.

$$LPHKZ_{p,q} = \sum_{l=2}^{\infty} \beta^l hyperpath_l(p, q) \quad (23)$$

where $hyperpath_l(p, q)$ is path of length l between the nodes p and q, defined as $s_1 s_2, \ldots s_l$, where $s_1 s_2, \ldots s_l$ are hyperlinks such that $s_1 \cap s_2 \neq \phi$. β, the damping factor such that $0 \leq \beta \leq 1$. In our experimentation, we set the maximum value for path of length l as 5. This concept draws from the idea of "six degrees of separation," suggesting that in real-world networks, most pairs of nodes are connected within a maximum path length of six. Many prior studies also adhere to a maximum path length of five.

We exhibit the superiority of the proposed approach over the existing versions of all these LP measures in the transformed graph empirically.

V. EXPERIMENTATION

A. DATASETS

We utilize the following six datasets for proving merit of the proposed approach over graphs.

- **NDC-classes-unique-hyperedges:** National Drug Code Directory (NDC) drugs dataset treats each class label (found in NDC-classes) as a node. A hyperlink is the collection of labels associated with a specific drug [41].
- **NDC-substances-unique-hyperedges:** This dataset also emerge from the National Drug Code Directory (NDC) drugs dataset, where every class substance (from NDC-substances) is denoted as a node. A hyperlink denotes the collection of substances associated with a specific drug [41].
- **email-Eu-unique-hyperedges (Email from European-research-institution):** Each node is an email address, and a hyperlink is formed with a group of nodes comprising the sender and all recipients associated with a particular email [42].
- **DAWN-unique-hyperedges (Drug abuse warning network (DAWN) drugs):** In this network, nodes represent drugs and hyperlink forms with a group of drugs employed by a patient [41].
- **tags-math-unique-hyperedges: Online question tags:** In this network, tags denote nodes and hyperlinks are the group of tags attached with a question on an online forum "https://math.stackexchange.com/".
- **tags-ask-ubuntu-unique-hyperedges: Online question tags:** This network is formed from an online forum of "https://askubuntu.com/". Nodes depict tags and hyperlinks, are group of tags attached with a question on an online forum.

Details about the datasets are provided in Table.4.

TABLE 4: Details of the datasets used in this work

Datasets	Number of Nodes	Number of Hyperlinks	Number of Pairwise links
NDC-classes-unique-hyperedges	1161	1088	6222
NDC-substances-unique-hyperedges	5311	9906	88268
email-Eu-unique-hyperedges	998	25027	29299
DAWN-unique-hyperedges	2558	141087	122963
tags-math-unique-hyperedges	1629	170476	91685
tags-ask-ubuntu-unique-hyperedges	3029	147222	132703

k-$fold$ cross-validation is used in this experimentation. The network is split up into k equal segments. Each time one part is taken as test set and all other k-1 parts combined is taken as train set. The proposed measures of $LPHRA$, $LPHCN$, $LPHJC$, $LPHAA$ and $LPHKZ$ are computed on the train set and evaluated against test set. The average of k iterations is reported. For comparison with existing measures, the train set is transformed into test set and the graph versions of the proposed measures are computed. The performance of these measures are evaluated on the same test hypergraph for uniformity.

We conducted our research on a computer system featuring an Intel(R) Core(TM) i7-8700 CPU, from 11th generation with a base clock speed of 3.20GHz, 6 cores, and 12 logical processors. The system had 16 GB of RAM and ran on Windows 10 Education. Our study was conducted using python, and implemented algorithms with libraries such as Networkx, Numpy, Pandas, Matplotlib, and Scikit-Learn.

B. EVALUATION METRICS

The performance of the suggested LP measures is commonly assessed using the following metrics [43].

- Area Under the Receiver Operating Characteristic Curve (AUROC)
- Precision
- F1-score

The calculation for AUROC, Precison, F1-score are based on True-Positive-Rate (TPR), True-Negative-Rate (TNR), False-Positive-Rate (FPR), False-Negative-Rate (FNR), which can easily computed from the confusion matrix.

where

- **True-Positive (TP):** The number of node pairings, whose links are both present in the test set and predicted by the LP measure.
- **True-Negative (TN):** The number of node pairings, having a link predicted by the LP measure but link is not existing in the test set.
- **False-Positive (FP):** The count of node pairings where LP measure does not anticipate the link, but connection is present in the test set.
- **False-Negative (FN):** The count of node pairs between which the LP measure does not predict a link, and the link doesn't exists in the test set.

Precision and F1-score are computed using Eq. 24 and Eq.26 respectively.

$$\text{Precision (PR)} = \frac{TP}{TP + FP} \tag{24}$$

$$\text{Recall (TPR)} = \frac{TP}{TP + FN} \tag{25}$$

$$\text{F1-score} = 2 \star \frac{\text{Precision} \star \text{Recall}}{\text{Precision} + \text{Recall}} \tag{26}$$

Precision (Eq.24) called link accuracy, defines how many of the positive predictions made are correct (true positives), whereas F1-score (Eq. 26) integrates recall and precision using a harmonic mean of two.

Area under ROC curve (AUROC) is a single point summary to specify the performance of a measure. AUROC has a range berween 0 and 1, 1 being the ideal value. Performance of any measure below 0.5 is taken as below random performance [44].

C. RESULTS

In this section, we discuss our findings on five proposed LP measures in hypergraphs such as $LPHRA$, $LPHCN$, $LPHJC$, $LPHAA$ and $LPHKZ$. We also compare against the corresponding LP measures in graphs.

Table. 5 presents AUROC scores across six networks. The bold font values highlights the top AUROC within LP graph measures, while red font values denote the highest across all

TABLE 5: Performance of LP measures in graph vs hypergraph in terms of AUROC

	LP in graphs					LP in Hypergraphs(LPH)				
Datasets	**RA**	**CN**	**JC**	**AA**	**KZ**	**LPHRA**	**LPHCN**	**LPHJC**	**LPHAA**	**LPHKZ**
NDC-classes	0.516	**0.722**	0.523	0.563	0.504	0.621	**0.789**	0.598	0.777	0.589
NDC-substances	0.523	0.598	0.501	**0.857**	0.503	0.955	0.629	0.729	**0.956**	0.596
email-Eu	0.525	0.501	0.502	**0.579**	0.508	**0.716**	0.603	0.698	0.606	0.644
DAWN	0.537	0.502	0.501	**0.632**	0.512	**0.852**	0.608	0.767	0.781	0.609
tags-math	0.535	0.501	0.499	**0.667**	0.515	**0.849**	0.696	0.754	0.691	0.651
tags-ask-ubuntu	0.572	0.502	0.499	**0.717**	0.501	**0.828**	0.712	0.717	0.762	0.696

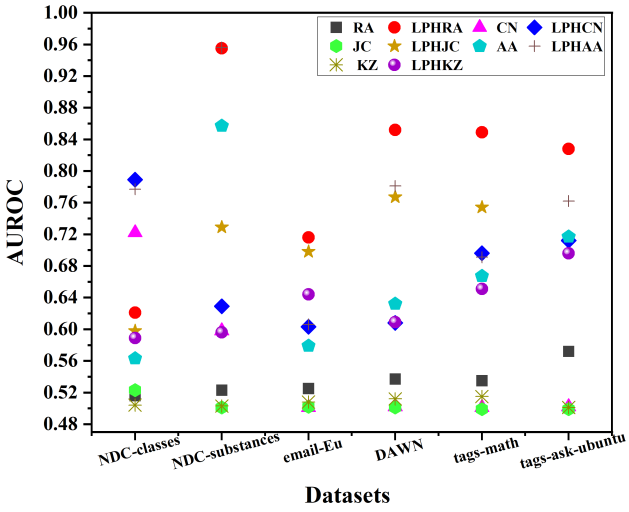

FIGURE 7: AUROC of predicting links in graphs vs hypergraphs

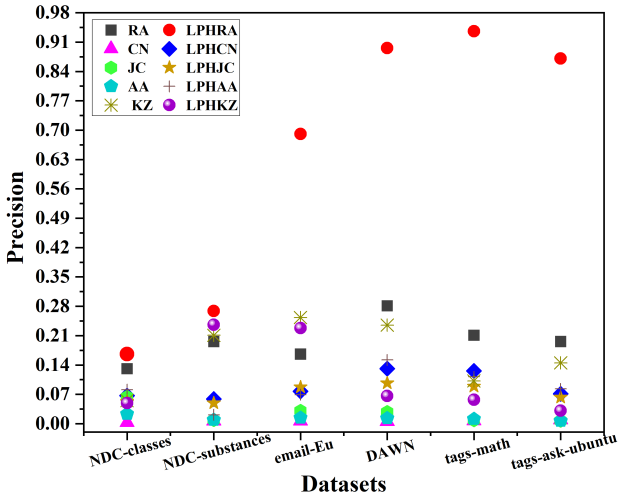

FIGURE 8: Precision of LP measures in graphs vs hypergraphs

evaluated metrics. Analysis of Table. 5 reveals AA's superior performance on five datasets: NDC-substances, email-Eu, DAWN, tags-math, and tags-ask-ubuntu. CN outshone others in the NDC-classes dataset. Among introduced LPH metrics, $LPHRA$ outpaced alternatives in LP for email-Eu, DAWN, tags-math, and tags-ask-ubuntu datasets. $LPHCN$ led in NDC-classes, with $LPHAA$ excelling in NDC-substances. Hypergraph adaptations of LP methods marked a 10% average enhancement in predictive accuracy over traditional graph-based predictions. The Fig. 7 showcases the AUROC for both graph and hypergraph LP metrics. Notably, the hypergraph iteration of RA, termed $LPHRA$, showcases a 27% boost over its RA counterpart. Likewise, $LPHCN$, $LPHJC$, $LPHAA$, and $LPHKZ$ have shown enhancements of 12%, 21%, 10%, and 13% respectively, when compared to their graph-based versions.

Table 6 details the precision scores for LP metrics. Within the NDC-classes dataset, RA leads in graph performance, closely followed by JC. However, in the hypergraph category, $LPHRA$ outshines its counterparts, with its precision notably marked in red. For the NDC-substances dataset, KZ stands out in graph metrics, while $LPHRA$, highlighted in red color font, achieves the highest precision among hypergraph measures, underscoring its effectiveness. In the email-EU dataset, KZ performs well in graph metrics, but LPHRA steals the show in hypergraph analysis, its excellence underscored in red color font. RA performs admirably in the DAWN dataset's graph metrics, with $LPHRA$ leading in hypergraph precision. In the tags-math dataset, RA tops the graph metrics, while $LPHRA$ distinguishes itself in the hypergraph domain. RA maintains strong performance in the graph metrics of the tags-ask dataset. $LPHRA$ often surpasses other hypergraph metrics, illustrating its capacity to capture complex connections. The success of hypergraph-based metrics, especially $LPHRA$, suggests their value for datasets with hypergraph structures.

Fig.8 illustrates the precision of LP metrics for both graph and hypergraph formats. LPHRA, the hypergraph adaptation of RA, shows a remarkable 44% improvement over its graph counterpart, RA. $LPHCN$ notes an 8% gain over CN. $LPHJC$ and $LPHAA$ mark advancements of 5% and 8%, respectively, against their graph-based iterations. $LPHKZ$, on the other hand, sees a modest 5% enhancement compared to KZ. In terms of precision, local similarity measures within the LPH framework excel over those based on global similarity, indicating a superior performance of hypergraph

TABLE 6: Performance of LP measures in graph vs hypergraph in terms of Precision

	LP in graphs					LP in Hypergraphs(LPH)				
Datasets	**RA**	**CN**	**JC**	**AA**	**KZ**	**LPHRA**	**LPHCN**	**LPHJC**	**LPHAA**	**LPHKZ**
NDC-classes	**0.131**	0.003	0.064	0.023	0.055	**0.166**	0.067	0.053	0.081	0.049
NDC-substances	0.196	0.006	0.009	0.009	**0.211**	**0.269**	0.059	0.049	0.021	0.236
email-Eu	0.166	0.006	0.031	0.014	**0.253**	**0.691**	0.077	0.087	0.069	0.228
DAWN	**0.281**	0.005	0.028	0.013	0.235	**0.896**	0.131	0.097	0.152	0.066
tags-math	**0.211**	0.007	0.008	0.011	0.102	**0.936**	0.126	0.089	0.112	0.057
tags-ask-ubuntu	**0.196**	0.009	0.007	0.007	0.145	**0.871**	0.071	0.063	0.083	0.031

TABLE 7: Performance of LP measures in graph vs hypergraph in terms of F1-score

	LP in graphs					LP in Hypergraphs (LPH)				
Datasets	**RA**	**CN**	**JC**	**AA**	**KZ**	**LPHRA**	**LPHCN**	**LPHJC**	**LPHAA**	**LPHKZ**
NDC-classes	**0.055**	0.006	0.054	0.036	0.016	**0.281**	0.046	0.167	0.044	0.019
NDC-substances	**0.077**	0.007	0.008	0.018	0.052	**0.444**	0.186	0.196	0.268	0.211
email-Eu	**0.081**	0.013	0.009	0.029	0.037	**0.605**	0.235	0.197	0.527	0.508
DAWN	**0.119**	0.011	0.016	0.025	0.046	**0.641**	0.227	0.257	0.546	0.518
tags-math	**0.108**	0.014	0.017	0.022	0.061	**0.666**	0.289	0.293	0.528	0.511
tags-ask-ubuntu	**0.142**	0.007	0.012	0.015	0.002	**0.645**	0.189	0.211	0.517	0.503

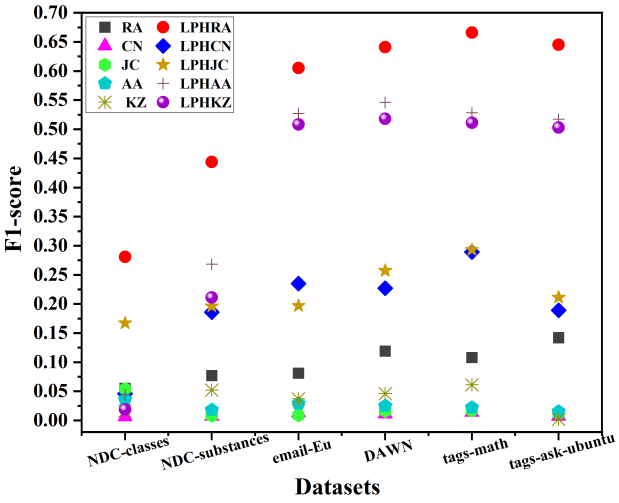

FIGURE 9: F1−score of LP measures in graphs vs hypergraphs

measures. LPH metrics have consistently surpassed their counterparts across all evaluated hyper-networks by an average margin of 0.7, signifying a significant decrease in false positives.

Table.7 showcases F1-score outcomes for LP and LPH metrics. Analysis reveals RA's impressive performance across various datasets, including NDC-substances, NDC-classes, email-Eu, DAWN, tags-math, and tags-ask Ubuntu. LPHRA, a proposed LPH metric, has shown superior LP capabilities across these six datasets. LPH metrics have proven to be significantly more effective than their graph-based equivalents, with a notable margin of 0.5. Fig.9 illustrates the superior performance of LPH metrics compared to graph-based metrics. LPH metrics excel in AUROC, accuracy, and F1-score evaluations. These improvements underscore the advantage of pairwise LP in hypergraphs over traditional graph conversion methods, alongside consistent efficacy across six distinct datasets. The superiority of LPH measures, with a margin of 0.5 in F1-scores over their graph counterparts, illustrates a significant leap in the precision-recall balance. This balance is crucial for effective LP, as it indicates a model's ability to identify true links without being overwhelmed by false positives. The LPH measures' dominance suggests that they are better tuned to capture the multidimensional relationships inherent in hypergraphs, which are often lost or oversimplified in graph conversions.

D. DISCUSSION

Analyzing five proposed LP strategies:$LPHRA$, $LPHCN$, $LPHJC$, $LPHAA$, $LPHKZ$—alongside their graph-based counterparts sheds light on the nuanced capabilities of hypergraph structures to model complex connections. The hypergraph variants consistently surpass graph-based methods in precision, accuracy, and F1-scores. Echoing findings from earlier studies on graph LP, metrics such as RA and AA excel due to their nuanced approach to calculating similarity, factoring in both the frequency of shared neighbors and the exclusivity of these connections. In contrast, CN and JC metrics, which merely tally shared neighbors without assessing their distinctiveness, lag in performance. The hypergraph method, particularly $LPHRA$, emerges as a superior predictor, outshining both other LPH metrics and traditional graph-based approaches. The global metric $LPHKZ$ underperforms in the AUROC metric, hindered by a hyperedge candidate set size capped at two, limiting its scope and slowing its computational pace due to the exhaustive consideration of graph topology.

While $LPHCN$ and $LPHJC$ falter in precision and F1-scores due to their constrained approach to common neighbors, the broader application of LPH metrics illuminates

their strength in capturing local interactions. This quality makes them potent for tasks like community detection and recommendation systems. The consistent 10% AUROC improvement with hypergraph models across various metrics underscores the inherent advantages of hypergraph representations. Despite the promising performance of hypergraph-based LP, challenges remain, including the need for more targeted global measures and improved computational efficiency and scalability in larger hypergraph contexts. Future research should address these limitations to enhance the utility and applicability of hypergraph-based LP methodologies.

VI. CONCLUSION

This investigation unveils a new strategy for predicting upcoming connections between entity pairs within complex hyper-networks. By harnessing the inherent topological features of hyper-networks, this strategy sidesteps the conventional need to convert these networks into simpler graph forms. It innovates by adapting five similarity-based LP metrics specifically for the nuanced environment of hyper-networks, tested across six standard complex hyper-network datasets. The newly developed LPH metrics have shown clear advantages over traditional LP methods.

In future, we intend to focus on extending more global LP measures to hypergraphs. We aim to investigate probabilistic approaches to gain deeper insights into the likelihood of future connections within a hypergraph framework. Further, the exploration of sophisticated machine-learning techniques, such as graph neural networks and deep learning, is on the agenda to boost the efficacy of LPH. The overarching aim is to enhance both the precision and utility of LP techniques within complex hyper-networks, thereby achieving a richer understanding of the dynamics within real-world complex systems. Similar to LP in graphs, our approach to LPH is currently limited to pairwise links due to concerns regarding time complexity. In consideration of privacy, we are only focusing on nodes and edges, neglecting node attributes and edge attributes, which represent certain drawbacks in our methodology. However, we plan to expand our research to encompass actual hyperlink prediction in the future. At present, node and edge attributes, as well as node centrality, are not being incorporated, but we aim to address these aspects in our future work.

VII. ABBREVIATIONS

The abbreviations used in this paper are given in Table. 8.

TABLE 8: Abbreviations

LP	Link Prediction
LPH	LP in Hyper-networks
CN	Common-Neighbors
JC	Jaccard-Coefficient
AA	Adamic-Adar
RA	Resource-Allocation
PA	Preferential-Attachment
KZ	Katz-Index
RWR	Random Walk With Restart
ACT	Average Commute Time
LRW	Local Random Walk
SRW	Superposed Random Walk
LPI	Local Path Index
LPM	Local Probabilistic Measure
PRM	Probabilistic Relational Measure
PERM	Probabilistic Entity Relationship Measure
HSM	Hierarchical Structure Measure
SBM	Stochastic Block Measure
SVD	Singular-Value-Decomposition
BS	Bayesian set
HPLSF	Hyperlink Prediction Using Latent Social Features
SHC	Spectral Hypergraph Clustering
MB	Matrix Boost
CMM	Coordinated Matrix Minimization
LPHCN	Link Prediction in Hypergraphs using Common Neighbor
LPHJC	Link Prediction in Hypergraphs using Jaccard Coefficient
LPHRA	Link Prediction in Hypergraphs using Resource Allocation
LPHAA	Link Prediction in Hypergraphs using Adamic Adar
LPHKZ	Link Prediction in Hypergraphs using Katz Index
TP	True-Positive
TN	True-Negative
FP	False-Positive
FN	False-Negative
AUROC	Area Under the Receiver Operating Characteristics

REFERENCES

[1] M. E. Newman, "Clustering and preferential attachment in growing networks," Physical review E, vol. 64, no. 2, p. 025102, 2001.

[2] S. H. Strogatz, "Exploring complex networks," nature, vol. 410, no. 6825, pp. 268–276, 2001.

[3] A. Frank, T. Király, and Z. Király, "On the orientation of graphs and hypergraphs," Discrete Applied Mathematics, vol. 131, no. 2, pp. 385–400, 2003.

[4] E. Estrada and J. A. Rodriguez-Velazquez, "Complex networks as hypergraphs," arXiv preprint physics/0505137, 2005.

[5] L. Getoor and C. P. Diehl, "Link mining: a survey," Acm Sigkdd Explorations Newsletter, vol. 7, no. 2, pp. 3–12, 2005.

[6] C. Wang, V. Satuluri, and S. Parthasarathy, "Local probabilistic models for link prediction," in Seventh IEEE international conference on data mining (ICDM 2007). IEEE, 2007, pp. 322–331.

[7] T. J. Lakshmi and S. D. Bhavani, "Link prediction approach to recommender systems," Computing, pp. 1–27, 2023.

[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, 2003, pp. 556–559.

[9] N. Yadati, V. Nitin, M. Nimishakavi, P. Yadav, A. Louis, and P. Talukdar, "Nhp: Neural hypergraph link prediction," in Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 2020, pp. 1705–1714.

[10] V. Martínez, F. Berzal, and J.-C. Cubero, "A survey of link prediction in complex networks," ACM computing surveys (CSUR), vol. 49, no. 4, pp. 1–33, 2016.

[11] 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.

[12] 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.

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

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

[15] A.-L. Barabâsi, 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, vol. 311, no. 3-4, pp. 590–614, 2002.

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

[17] 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). IEEE, 2006, pp. 613–622.

[18] W. Liu and L. Lü, "Link prediction based on local random walk," Europhysics Letters, vol. 89, no. 5, p. 58007, 2010.
[19] G. Jeh and J. Widom, "Simrank: a measure of structural-context similarity," in Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, 2002, pp. 538–543.
[20] M. A. Hasan and M. J. Zaki, "A survey of link prediction in social networks," Social network data analytics, pp. 243–275, 2011.
[21] A. K. Menon and C. Elkan, "Link prediction via matrix factorization," in Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2011, Athens, Greece, September 5-9, 2011, Proceedings, Part II 22. Springer, 2011, pp. 437–452.
[22] Z. Wu and Y. Chen, "Link prediction using matrix factorization with bagging," in 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS). IEEE, 2016, pp. 1–6.
[23] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," Computer networks and ISDN systems, vol. 30, no. 1-7, pp. 107–117, 1998.
[24] F. Chung and W. Zhao, "Pagerank and random walks on graphs," in Fete of combinatorics and computer science. Springer, 2010, pp. 43–62.
[25] 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, pp. 260–267, 2015.
[26] C. Chen and Y.-Y. Liu, "A survey on hyperlink prediction," IEEE Transactions on Neural Networks and Learning Systems, 2023.
[27] M. Zhang, Z. Cui, S. Jiang, and Y. Chen, "Beyond link prediction: Predicting hyperlinks in adjacency space," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, no. 1, 2018.
[28] T. Kumar, K. Darwin, S. Parthasarathy, and B. Ravindran, "Hpra: Hyperedge prediction using resource allocation," in Proceedings of the 12th ACM Conference on Web Science, 2020, pp. 135–143.
[29] D. Maurya and B. Ravindran, "Hyperedge prediction using tensor eigenvalue decomposition," Journal of the Indian Institute of Science, vol. 101, pp. 443–453, 2021.
[30] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, 2016, pp. 855–864.
[31] Z. Ghahramani and K. A. Heller, "Bayesian sets," Advances in neural information processing systems, vol. 18, 2005.
[32] Y. Xu, D. Rockmore, and A. M. Kleinbaum, "Hyperlink prediction in hypernetworks using latent social features," in Discovery Science: 16th International Conference, DS 2013, Singapore, October 6-9, 2013. Proceedings 16. Springer, 2013, pp. 324–339.
[33] M. Zhang, Z. Cui, T. Oyetunde, Y. Tang, and Y. Chen, "Recovering metabolic networks using a novel hyperlink prediction method," arXiv preprint arXiv:1610.06941, 2016.
[34] Q. Wang and G. Yan, "Hrwr: Predicting potential efficacious drug combination based on hypergraph random walk with restart," bioRxiv, pp. 2020–12, 2020.
[35] U. Chitra, "Random walks on hypergraphs with applications to disease-gene prioritization," Ph.D. dissertation, PhD thesis, Brown University, 2017.
[36] E. Nasiri, K. Berahmand, and Y. Li, "A new link prediction in multiplex networks using topologically biased random walks," Chaos, Solitons & Fractals, vol. 151, p. 111230, 2021.
[37] K. Berahmand, Y. Li, and Y. Xu, "A deep semi-supervised community detection based on point-wise mutual information," IEEE Transactions on Computational Social Systems, 2023.
[38] Shang, Yilun, "Non-linear consensus dynamics on temporal hypergraphs with random noisy higher-order interactions," Journal of Complex Networks, vol. 11, no. 2, p. cnad009, 2023.
[39] Shang, Yilun, "A system model of three body interactions in complex networks: consensus and conservation," Proceedings of the Royal Society A, vol. 478, no. 2258, p. 20210564, 2022.
[40] Shang, Yilun, "Consensus formation in networks with neighbor dependent synergy and observer effect," Communications in Nonlinear Science and Numerical Simulation, vol. 95, no. 2, p. 105632, 2021.
[41] M. T. Do, S.-e. Yoon, B. Hooi, and K. Shin, "Structural patterns and generative models of real-world hypergraphs," in Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 176–186.
[42] H. Yin, A. R. Benson, J. Leskovec, and D. F. Gleich, "Local higher-order graph clustering," in Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, 2017, pp. 555–564.
[43] Y. Yang, R. N. Lichtenwalter, and N. V. Chawla, "Evaluating link prediction methods," Knowledge and Information Systems, vol. 45, pp. 751–782, 2015.
[44] J. Han, M. Kamber, and J. Pei, "Data mining concepts and techniques third edition," University of Illinois at Urbana-Champaign Micheline Kamber Jian Pei Simon Fraser University, 2012.