CLPH: Link Prediction in Complex Hyper Networks Through Centrality Weighted Shared Connections

NANDINI, Y. V., LAKSHMI, T. Jaya <http://orcid.org/0000-0003-0183-4093> and ENDURI, Murali Krishna

SN Computer Science (2026) 7:119
https://doi.org/10.1007/s42979-025-04697-5

ORIGINAL RESEARCH

CLPH: Link Prediction in Complex Hyper Networks Through Centrality Weighted Shared Connections

Y. V. Nandini[1] · **T. Jaya Lakshmi**[2,3] · **Murali Krishna Enduri**[2]

Received: 20 March 2025 / Accepted: 23 December 2025

Abstract

Link Prediction (LP) aims to infer missing or future interactions in complex networks by exploiting structural patterns. Although widely applied in social, biological, and recommendation systems, traditional graph based LP methods rely solely on pairwise connections and therefore fail to capture the higher-order relationships that naturally arise in many real-world datasets. Hypergraphs offer a richer representation by allowing hyperedges to connect multiple nodes simultaneously. However, converting hypergraphs into simple graphs an approach commonly used in existing work collapses multi-node interactions and results in substantial information loss. Traditional LP metrics also treat all shared neighbors uniformly, despite the fact that shared neighbors may contribute differently to link formation depending on their structural importance or functional relevance. While centrality weighted LP extensions exist, they remain fundamentally restricted by graph structure and do not leverage higher-order dependencies. To address these limitations, we propose *CLPH*, a hypergraph based link prediction framework that incorporates hypercentrality to weight shared neighbors according to their structural influence. Experiments on four real-world hypergraphs demonstrate that CLPH achieves consistent improvements in AUPR, F1-score, and Precision. Notably, weighting shared neighbors using hypercentrality yields performance gains of 26%–68% compared to traditional centrality based weighting schemes.

Keywords Hypergraphs · Link prediction · Centralities · HyperCentralities

Introduction

Complex hypergraphs extend the traditional concept of complex graphs by allowing hyperedges to connect multiple nodes simultaneously, rather than being restricted to pairwise interactions. This multidimensional structure distinguishes hypergraphs from graphs, where edges represent simple relationships between two nodes. hypergraphs are more suited to modeling real-world systems that involve higher-order interactions, such as in social, biological, and technological domains [1]. For instance, in biological systems, protein-protein interaction networks often require more than just pairwise interactions to model multi-protein complexes. Similarly, in social networks, groups of individuals collaborating on tasks are better represented through hyperedges that link several nodes simultaneously, rather than through multiple pairwise edges. In complex hypergraphs [2], various tasks such as centrality measures, influence maximization, community detection, and link prediction present unique challenges due to the higher-order interactions. This work specifically focuses on link prediction and the use of centrality measures to improve its accuracy. Link Prediction [3] in complex hypergraphs differs significantly from traditional link prediction in graphs. In complex graphs, the focus is typically on predicting future pairwise edges between nodes based on their existing connections and shared neighbors. However, in hypergraphs, the task is more intricate,

✉ T. Jaya Lakshmi
j.tangirala@shu.ac.uk

Y. V. Nandini
yvnandini@vrsiddhartha.ac.in

Murali Krishna Enduri
muralikrishna.e@srmap.edu.in

[1] Department of Computer Science and Engineering, Siddhartha Academy of Higher Education, Deemed to be University, Vijayawada, Andhra Pradesh 520007, India

[2] Department of Computer Science and Engineering, SRM University-AP, Amaravati, Andhra Pradesh 522502, India

[3] Department of Computing, Sheffield Hallam University, Sheffield, UK

Published online: 19 January 2026

as hyperedges can involve multiple nodes, and the relationships between groups of nodes must be considered simultaneously. This introduces additional complexity because the formation of a hyperedge depends not just on individual pairwise interactions, but on the overall structure and connectivity of the hypergraphs. For example, in collaborative networks, while graph based link prediction might focus on predicting a new co-authorship between two researchers [4, 5], hypergraphs based link prediction aims to predict new collaborative groups involving multiple researchers. This shift from pairwise to higher-order interaction prediction is critical in real-world applications, such as identifying new scientific collaborations or forecasting interactions within protein complexes. Chen et al. [6] conducted a thorough survey on hyperlink prediction, categorizing the different methods used for link prediction in hypergraphs, which include similarity based approaches, probabilistic models, matrix based techniques, and deep learning methods. Among the various approaches to link prediction in hypergraphs, this study focuses on a select set of widely recognized similarity based methods. In particular, we emphasize shared neighbors and resource allocation, two prominent techniques that have been effectively extended to the hypergraphs paradigm. One of the main limitations of traditional link prediction methods in hypergraphs is that they often treat all shared neighbors as equally important, overlooking the varying influence of nodes within a hypergraphs. In reality, not all nodes contribute equally to the formation of new links. To address this issue, centrality measures, which capture the importance or influence of nodes within the network, can be incorporated into link prediction models to enhance their predictive power. Centralities such as degree centrality, closeness centrality, betweenness centrality, and clustering coefficient provide critical insights into a node's significance in the overall network structure. By integrating centrality scores into link prediction algorithms, we can better account for the differing influence of nodes, improving the accuracy of predictions. For instance, nodes with high degree centrality may be more likely to form new links, while nodes with high betweenness centrality may play crucial roles in connecting different subgroups within the hypergraphs. Centrality measures in hypergraphs are an extension of those in traditional graphs, but they need to account for the complexity of higher-order interactions. In graphs, centrality typically focuses on direct edges between nodes, but in hypergraphs, centrality must consider the role of nodes in hyperedges that involve multiple participants. For example: Hyperdegree centrality in a hypergraph measures the number of hyperedges a node is part of, as opposed to just the number of edges in a graph.

Hypercloseness centrality and hyperbetweenness centrality similarly extend their graph counterparts to account for the connectivity and influence of a node across groups of nodes, rather than simple pairwise connections. These adaptations are necessary because hypergraphs encode richer structural information than graphs, and nodes play more complex roles in the formation of new links. Using centrality measures to guide link prediction in hypergraphs provides a more nuanced approach to understanding and predicting interactions, especially in networks where the importance of nodes varies significantly across different contexts. In this work, we aim to leverage centrality measures to improve link prediction accuracy in complex hypergraphs, demonstrating their effectiveness across several real-world datasets. By comparing the performance of centrality based link prediction in hypergraphs with graph based approaches, we provide a clearer understanding of how centrality measures can enhance link prediction in networks with higher-order interactions. The study of [7] seeks to utilize the centrality scores of shared neighbors to enhance the accuracy of future link prediction. Prior to presenting the proposed approach, we review existing centrality measures used for link prediction in graphs [8] and discuss how these measures are adapted to the context of hypergraphs. Kshira Sagar Sahoo et al. [9] proposed an enhanced SDN security framework that detects DDoS attacks using an SVM model optimized with KPCA and a genetic algorithm. Their approach improves feature reduction, parameter tuning, and classification accuracy, making it suitable for real-time deployment within SDN controllers. S. Vimal et al. [10] propose a multiobjective Ant Colony Optimization and Double Q-learning–based energy-efficient clustering framework for IoT cognitive networks, improving network lifetime, throughput, and jamming prediction compared with conventional optimization methods. A. Rajagopal et al. [11] develop an optimal deep learning based UAV scene classification model combining residual network feature extraction with SGHS-optimized tuning and LVSVM classification, achieving superior accuracy. A. Rajagopal et al. [12] introduce an MBAS-ELM based distributed routing framework for LEO satellite networks that leverages traffic forecasting and mobile agents to achieve superior performance in delay, packet loss, and queuing efficiency compared with existing methods. G. Saranya et al. [13] proposed a DEL-CUBE framework using a hybrid Bald Eagle–Secretary Bird Optimization algorithm to achieve efficient and dynamic load balancing in cloud computing, that show significantly improved throughput and overall performance compared to existing methods.

The structure of this paper is organized as follows: Section Related Work reviews related work and discusses existing methods for link prediction and centrality measures in both graphs and hypergraphs. Section Proposed Work details the methodology, including the hypercentrality measures applied, the calculation of average hypercentrality, and the proposals of this paper. Section Experimentation describes the experimental setup, covering hypergraph sampling, datasets used, and presents the evaluation results. Sections Results and Discussion provides the results and an in-depth discussion of the findings and the results. Lastly, Section Conclusion and Future Work concludes the paper and suggests potential directions for future research.

Related Work

This section provides the necessary technical foundation, including key definitions and pertinent information relevant to this study. The formal definitions are presented as follows. Table.1 provides a summary of the notations used throughout this study.

Table 1 Notations employed throughout this study

Notation	Description
H	Hypergraphs
V	Set of Nodes
E	Set of Hyperedges
i,j,p,q	Nodes within the graphs and hypergraphs
$\Gamma(i), \Gamma(j)$	Neighbors of nodes i, j
$I_{\mathcal{H}}$	Incidence Matrix of hypergraph
d	Represents the distance between nodes
N	Represents total number of nodes within Graph and Hypergraph
$k(i)$	Degree of node i
$K(i)$	Number of triangles associated with node i
$he_1 he_2$	Hyperedges
s	size of hyperedge *he*
$LP(H)$	Link Prediction in Hypergraphs
$LP_{\mathcal{C}}$	Link Prediction based Centralities
$\mathcal{D}$	Degree
$\mathcal{CC}$	Clustering Coefficient
$\mathcal{B}$	Betweenness
$\mathcal{CL}$	Closeness Centrality
HSN	Shared Neighbor in Hypergraphs
HRA	Resource Allocation in Hypergraphs
$\mathcal{HC}$	HyperCentralities
$\mathcal{HD}$	HyperDegree
$\mathcal{HCC}$	HyperClustering Coefficient
$\mathcal{HB}$	HyperBetweenness
$\mathcal{HCL}$	HyperCloseness
$AC(H)$	Average HyperCentralities
$SN_{\mathcal{H}}\mathcal{C}$	Shared Neighbor based HyperCentralities
$RA_{\mathcal{H}}\mathcal{C}$	Resource Allocation based HyperCentralities
AUPR	Area Under the Precision-Recall Curve

Definition 1 Link Prediction in Hypergraphs (LP[H]): Given an undirected, unweighted hypergraph $H = (V, HE, t)$, where *V* is the set of vertices, *HE* is the set of hyperedges, and *t* is a time function on *E*, the link prediction in hypergraphs is to output a set of hyperedges that are not present in the hypergraphs $H[t_0, t_i]$, but are predicted to appear in $H[t_j]$ for $t_j > t_i > t_0$ [14].

Definition 2 Hyper Centralities $\mathcal{HC}$ **:** Given a hypergraph $H = (V, HE)$, where *V* denotes the set of vertices and *E* represents the set of hyperedges, the centrality in hypergraphs, referred to as hypercentrality, is defined as a function $C : V \to \mathbb{R}$, which assigns a real valued score *C*(*i*) to each node $i \in V$. This score quantifies the significance of *i* within the hypergraph structure, taking into account the node's participation in hyperedges and its connectivity across the hypergraph.

The score *C*(*i*) can be influenced by:

- The number of hyperedges that contain the node *i* (i.e., $|\{he \in HE : i \in he\}|$,
- The size of each hyperedge $he \in HE (i.e., |he|$ where $i \in he)$.
- The role of *i* in terms of its connections to other vertices within the hypergraph structure, either directly or through shared hyperedges.

Link Prediction in Hypergraphs

Link prediction in hypergraphs involves predicting the formation of future hyperedges among multiple nodes, where connections extend beyond the traditional pairwise relationships found in standard graphs. Unlike traditional graphs, hyperedges in hypergraphs link several nodes simultaneously, and the objective is to determine which groups of nodes are likely to establish new connections based on the structural properties of the hypergraph. Kumar et al. [15] address the challenge of hyperedge prediction, a complex task with applications in domains such as social networks and metabolic systems. The authors introduce HPRA (Hyperedge Prediction using Resource Allocation), a novel algorithm that predicts hyperedges of any size without the need for a predefined candidate set. Our work is largely inspired by the study conducted by [16], which focuses on local similarity measures. Local similarity based link prediction utilizes the immediate neighborhoods of nodes to compute similarity scores, typically based on shared connections. In this approach, the authors propose link prediction measures for hypergraphs directly, avoiding the conventional step of converting hypergraphs into pairwise graphs. The benefit of this method lies in its preservation of the intrinsic structure of hypergraphs, ensuring the original complexity and information integrity are maintained, unlike traditional approaches that simplify hypergraphs into standard graphs. The work defines two primary measures, HSN (Shared Neighbor in Hypergraph) and HRA (Resource Allocation in Hypergraph), both of which are elaborated upon below.

Shared Neighbor in Hypergraph: *HSN*

The authors in [16] extended the concept of shared neighbors to hypergraphs by computing the average of pairwise Shared Neighbor (*SN*) indices among the nodes within each hyperedge. The Link Prediction in hypergraphs using Shared Neighbors (*HSN*) is formally defined in Eq. 1 below.

$$HSN(i,j) = \frac{2}{s_{he_1}\, s_{he_2}} \sum_{r \in \Gamma(he_1) \cap \Gamma(he_2)} 1 \tag{1}$$

where, he_1 and he_2 are the hyperedges containing nodes *i* and *j*, s_{he} corresponds to the size of the hyperedge, $\Gamma(he)$ is the set of nodes incident to hyperedge *he*.

Resource Allocation in Hypergraph: *HRA*

The *HRA* method predicts pairwise links based on the principles of resource allocation, drawing inspiration from the work of [15]. Unlike in traditional graphs, hypergraphs permit nodes *i* and *j* to already belong to an existing hyperedge. As a result, resources at node *i* can be transferred to node *j* either directly or through shared neighbors. Consequently, the resource allocation between nodes *i* and *j* is determined in the below Eq. 2:

$$\begin{aligned} HRA(i,j) &= \sum_{i \neq j} \frac{1}{s_{he}-1}, \text{ if } i,j \in he \\ &= \sum_{r \in \Gamma(i) \cap \Gamma(j)} \frac{1}{k(r)} * \frac{1}{s_{he_1}-1} * \frac{1}{s_{he_2}-1}, \text{otherwise} \end{aligned} \tag{2}$$

where $\Gamma(i) \cap \Gamma(j)$ represents the shared neighbors of nodes *i* and *j*, *k*(*r*) is the degree of the shared neighbor *r*, s_{he_1} and s_{he_2} represent the sizes of the hyperedges containing *i* and *j*, respectively. The first term in Eq. 2 computes the resource transferred directly between *i* and *j* if they both belong to a shared hyperedge *he*. If they do not share the same hyperedges, this term evaluates to zero. The second term accounts for the resource transmitted through all shared neighbors between the two nodes.

Centralities and HyperCentralities

In graph theory, centrality measures are widely utilized to evaluate the importance or influence of individual nodes within a network. Traditional centrality metrics in graphs focus on pairwise relationships, measuring a node's significance based on its direct connections. Four of the most commonly applied centrality measures include degree centrality [17], clustering coefficient [18] [19], betweenness centrality [20], and closeness centrality [21]. These metrics provide insights into the structural role of nodes and their contributions to the overall dynamics of the network. Degree and clustering coefficient are classified as local centrality measures, as they consider only a node's immediate neighborhood and evaluate its importance based on its proximate connections. In contrast, betweenness and closeness centrality are prominent global measures that analyze the entire network to assess a node's significance, as discussed below.

- **Degree** $(\mathcal{D})$: The degree centrality of a node *i* is calculated as the fraction of nodes adjacent to *i* relative to the total possible connections within the network. Nodes with high degree centrality are commonly referred to as hub nodes, as they are highly connected and play a critical role in the network's structure.

$$\mathcal{D}(i) = \frac{k(i)}{N-1} \tag{3}$$

In Eq. 3, *N* represents the total number of nodes in the graph, and *k*(*i*) denotes the degree of node *i*, which refers to the number of direct connections that node *i* has within the network.

- **Clustering Coefficient** $(\mathcal{CC})$: The clusteringcoefficient of a node is defined as the ratio of the number of closed triangles (i.e., groups of three mutually connected nodes) within the node's local neighborhood to the total possible number of triangles that could exist in that neighborhood. This metric is also referred to as transitivity, reflecting the extent to which a node's neighbors are interconnected.

$$\mathcal{CC}(i) = \frac{2K(i)}{k(i)[k(i)-1]} \tag{4}$$

In Eq. 4, node *i* has a degree denoted by *k*(*i*), and the number of triangles associated with node *i* is represented as *K*(*i*).

- **Betweenness** ($\mathcal{B}$): Betweenness centrality of a node quantifies the extent to which the node lies on the shortest paths between other node pairs, reflecting its role in facilitating communication within the network.

$$\mathcal{B}(i) = \sum_{p \neq i \neq q} \frac{\sigma_{p,q}(i)}{\sigma_{p,q}} \tag{5}$$

In Eq. 5, $\sigma_{p,q}$ represents the total number of shortest paths between nodes *p* and *q*, while $\sigma_{p,q}(i)$ refers to the number of those shortest paths that pass through node *i*. This captures the extent to which node *i* acts as a bridge in the network, facilitating connections between other nodes.

- **Closeness** ($\mathcal{CL}$): Closeness centrality is a measure used to identify nodes that can effectively disseminate information across a network. The closeness centrality of a node i is calculated as the reciprocal of the average shortest path distance from node i to all other $N-1$ reachable nodes in the network. This measure highlights nodes that are well positioned to quickly interact with others within the network.

$$\mathcal{CL}(i) = \frac{N-1}{\sum_{i \neg j} d_{j,i}} \tag{6}$$

In Eq. 6, the shortest path distance between nodes *i* and *j* is represented by $d_{j,i}$. The node with the highest closeness centrality is the one that, on average, has the shortest path to all other nodes in the network, indicating its proximity and efficiency in reaching every other node.
Centrality measures have been extended to centrality based link prediction, where shared neighbors may not contribute equally to the formation of future links. Many researchers aims to evaluate the significance of shared neighbors in link prediction. Since centrality values reflect different forms of importance within a network, the centrality of shared neighbors influences the likelihood of link formation. The authors of [16] propose an approach for link prediction that relies on the average centrality of shared neighbors. This method, termed Link Prediciton on Centrality ($LP_{\mathcal{C}}$), computes a prediction score based on the similarity between nodes, factoring in the centrality scores of their shared neighbors. The method initially calculates various centrality scores for the shared neighbors and considers only those nodes whose scores exceed the average centrality value across the network. Let $\mathcal{C}(v)$ represent the centrality score of a node *i*, and A$\mathcal{C}(G)$ denote the average centrality value of the graph, which is computed using Eq. 7.

$$A\mathcal{C}(G) = \frac{\sum_{i \in V(G)} \mathcal{C}(i)}{N} \tag{7}$$

In Eq. 7, $\mathcal{C}(i)$ represents the centrality value of node *i*, and *N* denotes the total number of nodes in the graph *G*. The similarity between two vertices, based on the average centrality of the graph, is defined as shown in Eq. 8.

$$LP_{\mathcal{C}}(i,j) = \mid \{ r \mid r \in \Gamma(i) \cap \Gamma(j) \text{ and } \mathcal{C}(r) \geq A\mathcal{C}(G) \} \mid \tag{8}$$

In Eq. 8, $LP_{\mathcal{C}}(i,j)$ represents the similarity score between the nodepair *i* and *j*, calculated by identifying all shared neighbors and applying centrality scores to them. Only those shared neighbors whose centrality values exceed the average centrality of the graph are considered. Here, *r* denotes the shared neighbors between nodes *i* and *j*, while $\Gamma(i)$ and $\Gamma(j)$ represent the neighbors of nodes *i* and *j*, respectively. The average centrality ($A\mathcal{C}$) of the graph is as defined in Eq. 7. For instance, in link prediction *LP*, both Shared Neighbors (SN) and Resource Allocation (RA) can be incorporated using various centrality measures. If the centrality measure $\mathcal{C}$ represents degree centrality, the average degree centrality ($A\mathcal{D}$) can be calculated as defined in Eq. 7, facilitating the computation of similarity between two vertices based on the graph's average degree centrality. Moreover, the centrality measure $\mathcal{C}$ can be adapted to alternative centrality measures, such as betweenness centrality, closeness centrality, or clustering coefficient, depending on the context.

However, the centralities fail to capture more complex group interactions or multi-node connections that are prevalent in real-world networks. Hypercentrality measures in hypergraphs overcome this limitation by incorporating hyperedges, which connect multiple nodes simultaneously. These measures enable the identification of key nodes and relationships in the intricate topology of hypergraphs, where influence, connectivity, and centrality are defined by higher-order interactions rather than just pairwise links. Roy et al. [22] suggests using Shapley value-based centrality within a node centrality framework while maintaining the hypergraph structure. Li et al. [23] presents an innovative link prediction method for social networks through hypergraphs, which efficiently captures both pairwise and higher-order relationships, thereby improving the accuracy and effectiveness of link prediction tasks. Ihsan et al. [24] introduces entropy based centrality measures for hypergraphs, leveraging local similarities to determine centralities. Aksoy et al. [25] adapts graph metrics to s-metrics (higher-order hypergraph walks) in hypergraphs by utilizing their s-connected components. This approach involves first computing the s edge-adjacency matrix, which is then used to form the graph representation of the hypergraph. A few centrality measures for hypergraphs are briefly discussed below.

- **HyperDegree** $(H\mathcal{D})$: The hyperdegree of a node refers to the number of hyperedges that the node is involved in. Unlike traditional graphs, where edges connect only two nodes, hyperedges in hypergraphs can connect multiple nodes simultaneously. However, each hyperedge is counted only once for a node, irrespective of how many other nodes it connects. The core concept is that a node's significance or influence in a hypergraph is higher if it is part of more hyperedges, as this implies a wider range of connections and interactions within the network. The Degree Centrality of a node i in hypergraphs can be mathematically defined in Eq. 9:

$$\mathcal{HD}(i) = \sum_{j=1}^{m} IM_{i,he} \tag{9}$$

In this context, *IM* denotes the incidence matrix of the hypergraph, where $IM_{i,he}$ indicates the participation of node *i* in hyperedge *he*. The hyperdegree of node *i* is computed by summing the number of hyperedges in which node *i* is involved. This provides a measure of how connected node *i* is within the hypergraph.

- **HyperClustering Coefficient** $(H\mathcal{CC})$: In hypergraphs, the clusteringcoefficient is used to assess the likelihood of nodes forming tightly connected groups or clusters within the overall network structure [25]. Unlike traditional graphs, where clustering is measured by the probability that a node's neighbors are also directly connected, in hypergraphs, clustering evaluates the involvement of nodes in hyperedges that facilitate group interactions. This measure captures the probability that two nodes, which already share a hyperedge, are also linked by additional hyperedges. The formal mathematical expression for the clustering coefficient in hypergraphs is presented in Eq. 10.

$$\mathcal{HCC}(i) = \frac{\sum_{he_1,he_2} |he_1 \cap he_2|}{\binom{k(i)}{2}} \tag{10}$$

where $k(i)$ represents the set of hyperedges that contain node *i*, $\binom{k(i)}{2}$ denotes the total number of possible pairs of hyperedges involving i, and $|he_1 \cap he_2|$ refers to the size of the intersection between two hyperedges that both include *i*.

- **HyperBetweenness** $(H\mathcal{B})$: Betweenness Centrality in hypergraphs identifies nodes that serve as critical connectors within the network [25]. Nodes with a higher number of shortest paths passing through them are considered more influential, as they facilitate communication and interactions across various parts of the hypergraph. Betweenness centrality in hypergraphs can be calculated for both nodes and hyperedges; however, in this work, we focus exclusively on the centrality of nodes. The betweenness centrality for a node *i* (denoted as $H\mathcal{B}(i)$) is calculated using the following Eq. 11:

$$\mathcal{HB}(i) = \sum_{u \neq i \neq v} \frac{\sigma_{u,v}(i)}{\sigma_{u,v}} \tag{11}$$

In Eq. 11, $\sigma_{u,v}(i)$ represents the number of shortest paths from node *u* to node *v* that pass through node *i*, while $\sigma_{u,v}$ denotes the total number of shortest paths between nodes *u* and *v*. In traditional graphs, distance is defined by the number of edges in the shortest path connecting two nodes. However, in hypergraphs, hyperedges can connect multiple nodes simultaneously, altering the concept of distance. In its simplest form, the distance between two nodes in a hypergraph is defined as the number of hyperedges that must be traversed to connect them. If two nodes are part of the same hyperedge, the distance is 1. If they are not directly connected by a hyperedge, intermediate hyperedges must be traversed, increasing the distance.

- **HyperCloseness** $(H\mathcal{CL})$: Closeness centrality in hypergraphs measures how close a node is to all other nodes in the network [25]. Unlike traditional graphs, where distance is typically defined as the shortest path between two nodes, in hypergraphs, this concept is adapted to account for hyperedges that can simultaneously connect multiple nodes. Closeness centrality can be calculated for either nodes or edges within the hypergraph. If the edge parameter is set to True, it computes closeness centrality for edges; otherwise, it computes it for nodes. Additionally, the size of the hyperedges can be specified to adjust the calculation. The closeness centrality for a node i is computed using the Eq. 12:

$$\mathcal{HCL}(i) = \frac{N-1}{\sum_{i \neq j \in V} d(i,j)} \tag{12}$$

In this equation, $d(i, j)$ represents the distance between nodes i and j, while N denotes the total number of nodes in the hypergraph.

This study aims to adapt centrality measures to the more complex structure of hypergraphs, positing that centrality based approaches can yield deeper insights into connectivity patterns. Inspired by the work of [8], which introduced a novel link prediction method using similarity scores based on average centrality measures in traditional graphs, this research extends the methodology to hypergraphs. Metrics such as degree, clusteringcoefficient, betweenness, and closeness are modified to accommodate multi-node connections, offering a more comprehensive view of node influence in environments characterized by group interactions. By incorporating node centrality metrics into the link prediction framework, this method seeks to better capture the multifaceted relationships in hypergraphs, thereby improving predictive performance. The following section provides a detailed explanation of the proposed approach.

Proposed Work

The concept of link prediction for conventional graphs is rigorously defined in [3]. In this paper, we build upon that framework to tackle the challenge of link prediction and centralities in hypergraphs.

In this study, we introduce an innovative approach for predicting links in hypergraphs by utilizing node centralities,

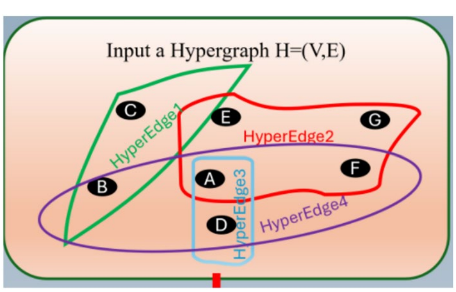


Fig. 1 Toy hypergraph with seven nodes and four hyperedges: $HE_1 = \{B, C, E\}$, $HE_2 = \{A, E, F, G\}$, $HE_3 = \{A, D\}$ and $HE_4 = \{A, B, D, F\}$

termed Link Prediction based HyperCentralities (*CLPH*). This method seeks to enhance the accuracy of link prediction by concentrating on shared nodes with greater influence within the network structure. To achieve this, we employ both the Shared Neighbor and Resource Allocation as link prediction metrics, both of which heavily rely on identifying shared neighbors. The *CLPH* algorithm is outlined in the following steps:

1. Computing the HyperCentralities for each node within the hypergraphs is a key step in this process. In this study, we employ four distinct centrality measures, which are detailed in Sect. 2.4.
2. Next, the average centralities of all nodes is computed, as illustrated in Eq. 13.

$$\mathcal{AC}(H) = \frac{\sum_{i \in V(H)} \mathcal{HC}(i)}{|V|} \tag{13}$$

In Eq. 13, $\mathcal{HC}(i)$ represents the HyperCentrality score of node i, computed using the hypergraph centrality measures outlined in Section Centralities and HyperCentralities. The generalized form of the Average Centrality score $\mathcal{AC}(H)$, as shown in Eq. 13, can be adapted based on the specific centrality measure utilized. For example, if the chosen HyperCentrality measure is hyperdegree, then $\mathcal{AD}(H)$ should be applied; if hyperbetweenness is used, then $\mathcal{AB}(H)$ replaces $\mathcal{AC}(H)$.

The criterion for selecting the average hypercentrality as the threshold is based on its role as a statistically representative baseline for the structural importance of nodes in a hypergraph. A node is considered influential only if its hypercentrality exceeds this mean value, indicating that its structural involvement is above the network's expected centrality level. This provides an objective and parameter free cutoff that naturally adapts to the centrality distribution of each dataset, ensuring consistent identification of meaningful shared neighbours. The use of the average also avoids distortions caused by hub nodes that would disproportionately affect maximum based thresholds. Applying this criterion has been empirically shown to suppress low-influence neighbours and improve the reliability of the computed link prediction scores.

3. To predict a potential link between two nodes, i and j, the method first identifies their shared neighbors within the hypergraphs. In the subsequent step, only shared neighbors whose HyperCentrality scores exceed the average centrality are considered for calculating the link prediction scores. These selected shared neighbors are then used to compute the link prediction scores. In this work,

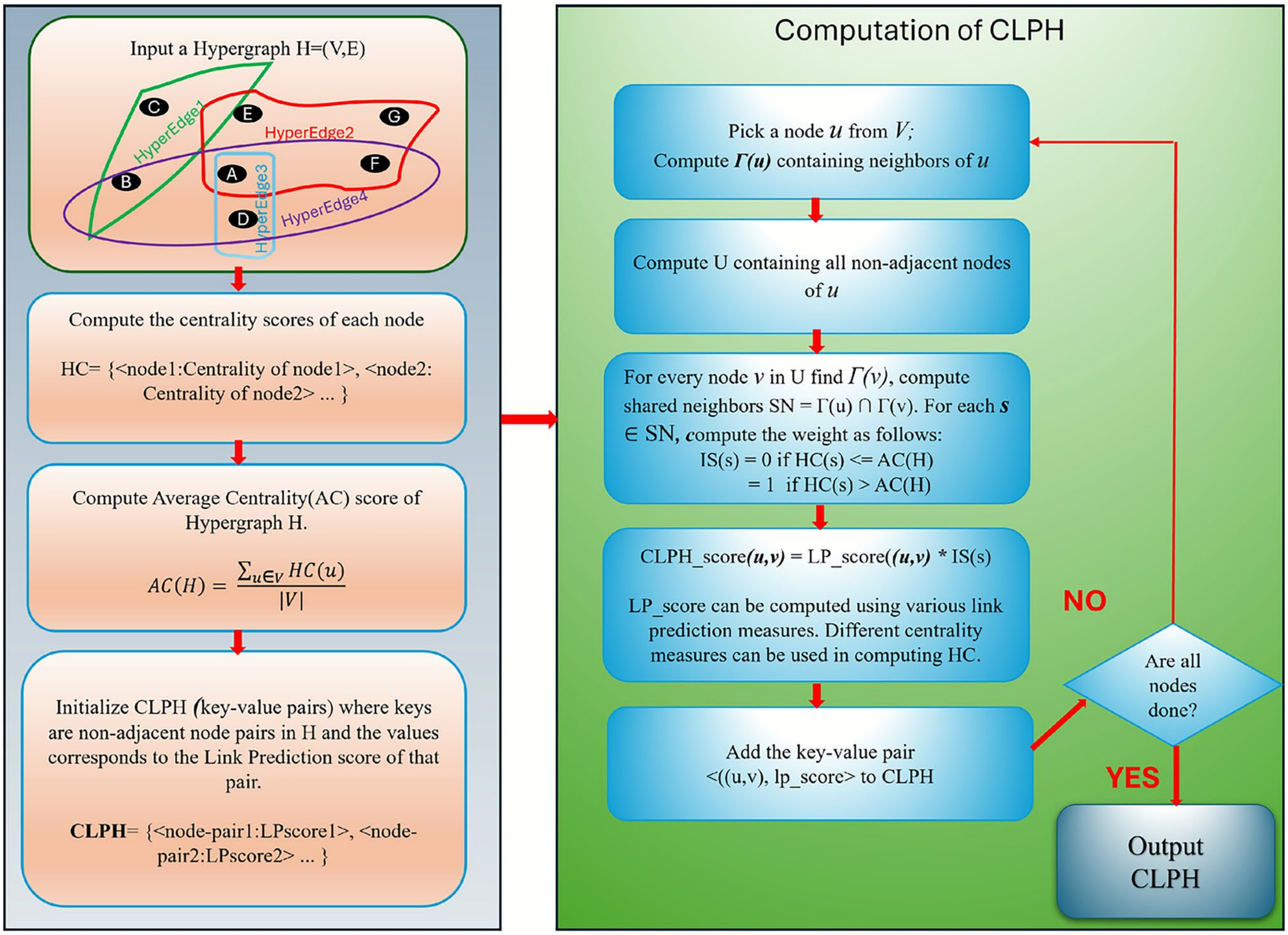


Fig. 2 Link Prediction in hypergraphs through Centrality Weighted Shared Connections

Table 2 The proposed link prediction measures based on HyperCentralities ($CLPH(i, j)$)

	Average HyperCentralities ($\mathcal{AC}$)	$CLPH_{\mathcal{HC}}(i,j)$
HyperDegree ($\mathcal{HD}$)	$A\mathcal{HD}(H) = \frac{\sum_{i \in V(H)} \mathcal{HD}(i)}{\lvert V \rvert}$	$SN_{\mathcal{HD}}(i,j) = \lvert\{r \mid r \in \Gamma(i) \cap \Gamma(j) \text{ and } \mathcal{HD}(r) \geq A\mathcal{HD}(H)\}\rvert$
HyperClustering Coefficient ($\mathcal{HCC}$)	$A\mathcal{HCC}(H) = \frac{\sum_{i \in V(H)} \mathcal{HCC}(i)}{\lvert V \rvert}$	$SN_{\mathcal{HCC}}(i,j) = \lvert\{r \mid r \in \Gamma(i) \cap \Gamma(j) \text{ and } \mathcal{HCC}(r) \geq A\mathcal{HCC}(H)\}\rvert$
HyperBetweenness ($\mathcal{HB}$)	$A\mathcal{HB}(H) = \frac{\sum_{i \in V(H)} \mathcal{HB}(i)}{\lvert V \rvert}$	$SN_{\mathcal{HB}}(i,j) = \lvert\{r \mid r \in \Gamma(i) \cap \Gamma(j) \text{ and } \mathcal{HB}(r) \geq A\mathcal{HB}(H)\}\rvert$
HyperCloseness ($\mathcal{HCL}$)	$A\mathcal{HCL}(H) = \frac{\sum_{i \in V(H)} \mathcal{HCL}(i)}{\lvert V \rvert}$	$SN_{\mathcal{HCL}}(i,j) = \lvert\{r \mid r \in \Gamma(i) \cap \Gamma(j) \text{ and } \mathcal{HCL}(r) \geq A\mathcal{HCL}(H)\}\rvert$

two specific link prediction measures are employed, as detailed in Section Link prediction in Hypergraphs.

For the toy hypergraph shown in Fig. 1, hypercentrality is computed using hyperdegree, yielding $HD(A) = 3$, $HD(B) = 2$, $HD(C) = 1$, $HD(D) = 2, HD(E) = 2, HD(F) = 2, HD(G) = 1$, with average hyperdegree

$$AD(H) = \frac{13}{7} \approx 1.857,$$

so the nodes satisfying the threshold $(HD(v) \geq AD(H)$. For example, for the non-adjacent nodepair (*A*, *C*) we obtain $\Gamma(A) = \{B, D, E, F, G\}$ and $\Gamma(C) = \{B, E\}$, hence $SN_H D(A, C) = (B, E)$. Rather than treating these neighbors uniformly, we incorporate their hyperdegree centrality: both B and E have $HD = 2$, which exceeds *AD*(*H*), and therefore (*A*, *C*) passes the centrality threshold and is retained as a candidate for link prediction. In general, for each non-adjacent nodepair, only those shared neighbors satisfying $(HC(r) \geq AD(H)$ contribute to the score; pairs with no qualifying neighbors are discarded. This selective inclusion focuses the predictor on structurally meaningful candidate pairs and improves prediction quality relative to methods that treat all non-adjacent nodepairs equally.

The computational procedure is depicted in Fig. 2

HyperCentralities

In this research, we employ several hypergraph specific centralities, such as hyperdegree, hyperclustering coefficient, hyperbetweenness, and hypercloseness, in place of $\mathcal{HC}$. The precise mathematical formulations for these hypercentralities are outlined in Table 2, demonstrating their adaptation to hypergraphs for accurately capturing node influence.

Link Prediction based HyperCentralities in Hypergraphs

(a) **Shared Neighbor based HyperCentralities** $(SN_{\mathcal{H}}\mathcal{C})$: To calculate the similarity scores between non-adjacent nodepairs (*i*, *j*) in hypergraphs using average centrality, we extend the notion of shared neighbors to incorporate hyperedges. The Shared Neighbor based HyperCentralities for hypergraphs, denoted as $SN_{\mathcal{HC}}$, is formally defined in Eq. 14:

$$SN_{\mathcal{HC}}(i,j) = \frac{2}{s_{he_1}\, s_{he_2}} \sum_{\substack{r \in \Gamma(he_1) \cap \Gamma(he_2) \\ HC(r) \geq AC(H)}} HC(r) \tag{14}$$

Here, he_1 and he_2 represent hyperedges, with s_{he} indicating the size of hyperedge *he*. The term *r* refers to a shared neighbor shared between the hyperedges, while $\mathcal{HC}(r)$ denotes the hypercentrality score of the shared neighbor *r*. The average hypercentrality, $AC(H)$, is defined in Eq. 13.

The given Eq. 14 determines the link prediction score by evaluating the number of shared neighbors whose centrality values are equal to or exceed the average centrality of the hypergraph. The hypercentrality measure, $\mathcal{HC}$, can be adapted to other hypergraph centrality metrics, such as the hyperdegree ($\mathcal{HD}$), hyperclustering coefficient ($\mathcal{HCC}$), hyperbetweenness centrality ($\mathcal{HB}$), and hypercloseness centrality ($\mathcal{HCL}$), as presented in Table 2. These adaptations result in the computation of $CN_{\mathcal{H}}\mathcal{C}$.

(b) **Resource Allocation based HyperCentralities** $(RA_{\mathcal{H}}\mathcal{C})$: In traditional graphs, the Resource Allocation (*RA*) Index operates on the principle of distributing resources between two nodes through their shared neighbors. In hypergraphs, the Resource Allocation Index is adapted to accommodate the complexity of hyperedges, which can connect multiple nodes simultaneously. Instead of focusing on pairwise neighbors, the RA score in hypergraphs evaluates resource distribution through shared hyperedges. This approach quantifies the potential for "resource" transfer between two nodes based on their shared hyperedges, assigning greater importance to smaller hyperedges where connections are more concentrated. The Resource Allocation based HyperCentralities for hypergraphs is formally defined in Eq. 15:

$$\begin{aligned} RA_{\mathcal{H}}\mathcal{C}(i,j) &= \sum_{i \neq j} \frac{1}{s_{he}-1}, \text{ if } i,j \in he \\ &= \sum_{\substack{r \in \Gamma(i) \cap \Gamma(j) \\ \text{and } \mathcal{HC}(r) \geq AC(H)}} \frac{1}{k(r)} * \frac{1}{s_{he_1}-1} * \frac{1}{s_{he_2}-1}, \text{otherwise} \end{aligned} \tag{15}$$

where $\Gamma(i) \cap \Gamma(j)$ represents the shared neighbors of nodes *i* and *j*, $\mathcal{HC}(r)$ is the hypercentrality score of the shared neighbor *r*, $AC(H)$ is the average hypercentrality of the hypergraph *H*, *k*(*r*) is the degree of the shared neighbor *r*, s_{he_1} and s_{he_2} represent the sizes of the hyperedges containing *i* and *j*, respectively. $RA_{\mathcal{H}}\mathcal{C}$ can also be formulated using different HyperCentrality measures, analogous to those presented in Table.2.

The effectiveness of the proposed method is evaluated through experimental testing on four distinct hypergraphs. A detailed account of this evaluation is provided in the following section. The proposed method is summarised in Algorithm 1.

Initialization: $CLPHScores = \phi$ // $LP_{\mathcal{HC}}Scores$ holds key-value pairs, where the keys represent non-adjacent nodepairs (i, j), and the values correspond to their respective $CLPH$ scores.
for every vertex i **in** V **do**
  Calculate $\mathcal{HC}(i)$ // HyperCentralities for node i
end for
Compute $\mathcal{AC}(H) = \frac{\sum_{i \in V} \mathcal{HC}(i)}{|V|}$ // AverageCentralities of the hypergraph $\mathcal{AC}(H)$
for every vertex i in V **do**
  Compute $\Gamma(i)$ // Set of neighbors of node i
  for every vertex j in V **do**
    if $(i, j) \notin E$ **then** // Non-adjacent nodepair
      Compute $\Gamma(j)$ // Set of neighbors of node j
      Compute $r = \Gamma(i) \cap \Gamma(j)$ // Shared neighbors of i and j
      for every r **do**
        if $\mathcal{HC}(r) \leq \mathcal{AC}(H)$ **then**
          Remove r // Retain only the significant shared neighbors based on hypercentrality, and similarly, for resource allocation, compute by adding the degrees of the shared neighbors to facilitate resource transfer.
        end if
      end for
      Step 1: Shared Neighbor based Link Prediction:
      Calculate $SN_{\mathcal{HC}}$ from Eq.14 // Shared Neighbor based HyperCentralities
      Step 2: Resource Allocation based Link Prediction:
      Calculate $RA_{\mathcal{HC}}$ from Eq.15 // Resource Allocation based Hypercentralities
      Add $((i, j) : (SN_{\mathcal{HC}}Scores, RA_{\mathcal{HC}}Scores)) to LP_{\mathcal{HC}}Scores$ // Store both scores
    end if
  end for
end forreturn $CLPH$ Scores

The *CLPH* algorithm proceeds through the following stages:

(a) **Computation of node HyperCentralities.** For each node in the hypergraph, the corresponding HyperCentrality score $\mathcal{HC}(i)$ is computed. In this study, we employ four distinct hypergraph adapted centrality measures like hyperdegree, hyperclustering coefficient, hyperbetweenness, and hypercloseness as described in Section Centralities and HyperCentralities.

(b) **Derivation of the average HyperCentrality baseline.** Once individual node centralities are obtained, the average HyperCentrality of the hypergraph is computed using Eq. 16:

$$\mathcal{AC}(H) = \frac{\sum_{i \in V} \mathcal{HC}(i)}{|V|}. \tag{16}$$

Here, $\mathcal{HC}(i)$ denotes the HyperCentrality of node *i*, as defined by the selected measure. The expression in Eq. 16 is general and adapts to any centrality variant used. For example, when the chosen HyperCentrality corresponds to hyperdegree, hyperbetweenness, hyperclustering coefficient, or hypercloseness, the corresponding average values $\mathcal{AHD}(H)$, $\mathcal{AHB}(H)$, $\mathcal{AHCC}(H)$, and $\mathcal{AHCL}(H)$ are employed respectively.

c) **Identification and filtering of structurally significant shared neighbours.** For each non-adjacent pair of nodes (i, j), their shared neighbours are identified. Only those neighbours whose HyperCentrality values exceed the average centrality baseline $\mathcal{AC}(H)$ are retained for subsequent computation. These structurally

Table 3 Datasets containing nodes and hyperedges after sampling

Datasets	#Nodes	#Hyperedges
NDC-classes	640	411
email-Eu	745	4868
cat-edge-geometry-questions	450	247
hyperedges-contact-high-school	317	781

influential shared neighbours are then used to compute the link prediction scores. In this work, two centrality weighted link prediction indices like Shared Neighbour based and Resource Allocation based are utilized, as detailed in Section Proposed Work.***Computational Complexity.*** The computational cost of the proposed *CLPH* algorithm can be analysed by considering the major operations involved in the procedure. Computing the Hypercentrality score for all nodepairs requires $O(|V| \cdot C_{\mathcal{HC}})$ time, where $C_{\mathcal{HC}}$ denotes the complexity of the selected centrality measure. The identification of all non-adjacent node pairs involves a worst-case cost of $O(|V|^2)$. For each such pair (i, j), determining the shared neighbours entails extracting the neighbour sets $\Gamma(i)$ and $\Gamma(j)$ and computing their intersection, which requires $O(k(i) + k(j))$ time. The subsequent filtering of shared neighbours based on the average Hypercentrality threshold incurs an additional $O(|\Gamma(i) \cap \Gamma(j)|)$ cost. The computation of the Shared Neighbour score $SN_{\mathcal{HC}}$ and the Resource Allocation score $RA_{\mathcal{HC}}$ also requires $O(|\Gamma(i) \cap \Gamma(j)|)$ time for each pair.

Combining these factors, the overall worst-case time complexity of the algorithm is given by

$$O(|V|^2 \cdot \Delta),$$

where Δ is the maximum node degree in the hypergraph. This complexity is consistent with neighbourhood based link prediction methods and reflects the inherent cost of evaluating pairwise interactions in hypergraph structures.

Experimentation

We utilized four datasets to demonstrate the effectiveness of the proposed approach, each dataset sampled from hypergraphs available in the ARB repository "https://www.cs.cornell.edu/~arb/data/email-Eu/".

- **National Drug Code Directory** *NDC* [26]: In the NDC-classes dataset, nodes represent class labels assigned to drugs, with each node corresponding to a particular label linked to a drug. Hyperedges are formed by simplices, where each simplex represents a group of nodes (class labels) interconnected by a drug.
- **email-Eu:** In the email-Eu dataset, nodes correspond to email addresses within a European research institution, with a hyperedge formed by grouping the sender and all recipients involved in a particular email. "https://www.cs.cornell.edu/~arb/data/email-Eu/"
- **cat-edge-geometry-questions:** In this hypergraph dataset, nodes represent individual geometry related questions, while hyperedges are formed by grouping questions that are conceptually linked, indicating they share shared topics and connect multiple questions (nodes) within a single hyperedge.
- **hyperegdes-contact-high-school** [27]: In this dataset, nodes represent individuals at the high school who interacted with each other, while hyperedges correspond to the maximal cliques of these interactions, represented as simplices, where each hyperedge links all individuals who engaged in mutual interactions (see Table 3).

Due to the large size of the hypergraphs, we applied a sampling approach based on hyperedge distribution to reduce their scale. The hyperedge distribution function indicates the number of hyperedges of each possible size (cardinality), effectively describing how many hyperedges contain a specified number of nodes. In most datasets, pairwise hyperedges (size 2) are predominant, while larger hyperedges (sizes 3, 4, etc.) occur with decreasing frequency. This distribution reveals the prevalence of pairwise interactions in hypergraphs. Our research, while focused on hypergraphs where hyperedges can link multiple no des, primarily considers non-adjacent node pairs (hyperedge size 2). This focus is supported by the observation that approximately 60-70% of the datasets consist of hyperedges involving exactly two nodes, making it efficient and insightful to start with these smaller hyperedges. The sampling process organizes hyperedges by size to allow for proportional sampling. From each size group, a user-defined fraction of hyperedges (e.g., 50%) is selected, with at least one hyperedge retained in each group to ensure continuity. Isolated nodes, which no longer connect to any hyperedge, are removed. This approach effectively reduces hypergraph size while preserving critical node relationships, facilitating analysis within time and memory constraints. Grouping hyperedges by size and sampling proportionally ensures that the sampled hypergraphs maintain a distribution akin to the original data. This study was conducted on a system equipped with an 11th generation Intel(R) Core(TM) i7-8700 CPU, six cores, twelve logical processors, and a base clock of 3.20 GHz, running Windows 10 Education with 16 GB RAM. Python was used for the analysis.

Evaluation Metrices

Although various evaluation metrics exist for link prediction, our study focuses on AUPR, precision, and F1-score for several key reasons. AUPR is particularly effective in handling class imbalance, a common scenario in link prediction tasks where actual links are sparse, which makes sure for predicting important links. Precision ensures that the predicted links are highly accurate, which is crucial for practical applications such as biological networks and collaboration networks, where incorrect predictions can lead to misleading outcomes whereas F1-score strikes a balance between precision and recall, ensuring that the model does not prioritize one at the expense of the other which assess overall performance in link prediction.

- **AUPR** [28]: AUPR refers to the area under the precision-recall curve, where precision is plotted on the y-axis and recall on the x-axis across various threshold values. It offers a comprehensive summary of the trade-off between precision and recall, highlighting the model's ability to correctly classify positive instances. In the context of link prediction, particularly in hypergraphs, datasets are frequently imbalanced, with far fewer links (positive class) compared to non-links (negative class). AUPR directly addresses this by focusing on how well the model identifies true links, while disregarding irrelevant connections that could skew other metrics. For link prediction tasks involving centrality measures, AUPR ensures that the model is evaluated on its ability to correctly identify rare but significant links. It provides a more accurate evaluation than metrics like accuracy, which can be misleading in skewed datasets.
- **Precision** [29]: Precision quantifies the proportion of correctly predicted links (true positives) among all predicted links (true positives + false positives). Mathematically, it is expressed in Eq. 17:

$$Precision = \frac{TP}{TP + FP} \tag{17}$$

where TP denotes true positives and FP represents false positives. Precision is critical in scenarios where minimizing false positives is a priority. In hypergraphs, predicting non-existent links can disrupt the understanding of the network structure, making precision a key metric in evaluating the reliability of a model's predictions. Precision is particularly important in hypergraphs with higher-order interactions, where the cost of incorrect predictions can be significant. For instance, in protein-protein interaction networks, inaccurately predicting links can result in false biological inferences, affecting critical research outcomes.

- **F1-score** [30]: The F1-score is the harmonic mean of precision and recall, providing a balanced evaluation of a model's performance. It is calculated in Eq. 18 as:

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{18}$$

Table 4 Performance of centrality based LP in graph Vs hypergrpah modes. $\mathcal{D}$ and $\mathcal{HD}$ represent Degree Centrality in the graph and hypergraph models respectively. *SN* and *RA* signify the Link Prediction (LP) measures of Shared Neighbor and Resource Allocation

Evaluation Metrics	AUPR				F1-score				Precision			
LP Measures	$SN_{\mathcal{D}}$	$SN_{\mathcal{HD}}$	$RA_{\mathcal{D}}$	$RA_{\mathcal{HD}}$	$SN_{\mathcal{D}}$	$SN_{\mathcal{HD}}$	$RA_{\mathcal{D}}$	$RA_{\mathcal{HD}}$	$SN_{\mathcal{D}}$	$SN_{\mathcal{HD}}$	$RA_{\mathcal{D}}$	$RA_{\mathcal{HD}}$
NDC-classes	0.019	**0.024**	0.017	0.021	0.162	**0.861**	0.184	0.786	0.167	0.462	0.215	**0.513**
email-Eu	0.072	**0.121**	0.067	0.112	0.103	**0.838**	0.099	0.753	0.211	0.521	0.282	**0.616**
c-e-g-q	0.029	0.286	0.067	**0.313**	0.231	0.654	0.261	**0.712**	0.115	**0.519**	0.156	0.502
h-c-h-s	0.055	0.077	0.065	**0.086**	0.258	0.868	0.229	**0.872**	0.131	0.498	0.149	**0.512**

This metric combines both precision and recall to offer a single, comprehensive measure. The F1-score effectively balances the trade-off between precision (minimizing false positives) and recall (minimizing false negatives), making it especially useful in situations where both types of errors are critical. In link prediction tasks, the F1-score ensures that the model is not only avoiding false positives (precision) but also capturing as many true links as possible (recall). In hypergraphs, it is insufficient for the model to only have high precision (predicting a few but accurate links). It must also achieve high recall by identifying as many true links as possible. The F1-score ensures that the model maintains this balance, which is essential in domains such as scientific collaboration or biological networks, where missing a true link can be as detrimental as predicting a false one.

Results

The simulations for the proposed hypergraph based link prediction models were performed using the top 20,000 node-pairs, with results averaged over 10 data points to ensure greater accuracy and reliability. The evaluated HyperCentrality measures include Shared Neighbor based Hyper-Degree ($SN_{\mathcal{HD}}$), Shared Neighbor based HyperClustering Coefficient ($SN_{\mathcal{HCC}}$), Shared Neighbor based HyperBetweenness ($SN_{\mathcal{HB}}$), and Shared Neighbor based Hyper-Closeness ($SN_{\mathcal{HCL}}$), along with their Resource Allocation based counterparts ($RA_{\mathcal{HD}}$, $RA_{\mathcal{HCC}}$, $RA_{\mathcal{HB}}$, and $RA_{\mathcal{HCL}}$). These hypergraph measures were compared to graph based similarity centralities, including $SN_{\mathcal{D}}$, $SN_{\mathcal{CC}}$, $SN_{\mathcal{B}}$, $SN_{\mathcal{CL}}$, and $RA_{\mathcal{D}}$, $RA_{\mathcal{CC}}$, $RA_{\mathcal{B}}$, $RA_{\mathcal{CL}}$. Performance was assessed using three key metrics: Area under Precision-Recall Curve (AUPR), F1 score, and Precision, as summarized in Tables 4, 5, 6 and 7.

- **AUPR Analysis**
  For the NDC-classes dataset which is in Table.4, the highest AUPR of 0.024 is achieved by $SN_{\mathcal{HD}}$, representing a 26.3% improvement over the graph based $SN_{\mathcal{D}}$ (0.019). This result underscores the effectiveness of hyperdegree centrality in capturing multi-node interactions and dense connectivity patterns characteristic of hypergraphs. Graph based measures, such as $SN_{\mathcal{D}}$, rely on pairwise relationships and fail to account for the broader structural participation of nodes within hyperedges. In the email-Eu dataset, $SN_{\mathcal{HD}}$ achieves the highest AUPR of 0.121, which is 68% higher than $SN_{\mathcal{D}}$ (0.072). The dataset reflects communication networks where nodes frequently engage in multiple hyperedges. Hyperdegree centrality excels in identifying nodes with diverse and frequent interactions, making it more robust in predicting future links. For the cat-edge-geometry-questions dataset, $RA_{\mathcal{HD}}$ achieves the best AUPR score of 0.313, which is 9.4% higher than $SN_{\mathcal{HD}}$ (0.286). The Resource Allocation based hyperdegree measure effectively models the direct connectivity and interaction dynamics within geometrically organized clusters, demonstrating its superiority in this scenario. In the hyperedges-contact-high-school dataset, $RA_{\mathcal{HD}}$ leads with an AUPR of 0.086, surpassing $SN_{\mathcal{HD}}$ (0.077) by 11.7%. This dataset represents group based social interactions where resource allocation methods capture the probabilistic formation of links within closely-knit communities more effectively than shared neighbor-based methods.
- **F1-Score Analysis**
  For the NDC-classes dataset which is in Table 4, $SN_{\mathcal{HD}}$ achieves the highest F1-score of 0.861, which is 9.5% higher than $RA_{\mathcal{HD}}$ (0.786). The F1-score balances precision and recall, highlighting $SN_{\mathcal{HD}}$'s ability to identify future links accurately while minimizing false positives. Its superior performance is attributed to the hypergraph's inherent structure, where nodes exhibit dense connectivity within hyperedges. In the email-Eu dataset, $SN_{\mathcal{HD}}$ achieves the highest F1-score of 0.838, surpassing $RA_{\mathcal{HD}}$ (0.753) by 11.3%. This dataset, characterized by frequent multi-node interactions, benefits from hyperdegree centrality's focus on nodes central to existing hyperedges, resulting in more reliable link predictions. For the cat-edge-geometry-questions dataset, $SN_{\mathcal{HD}}$ leads with an F1-score of 0.654, which is 10.5% higher than $SN_{\mathcal{HD}}$ (0.591). This improvement demonstrates $SN_{\mathcal{HD}}$'s ability to capture dense, localized group interactions within geometrically structured hyperedges, outperforming other methods. In the hyperedges-contact-high-school dataset, $RA_{\mathcal{HD}}$ records the highest F1-score of 0.872, marginally outperforming $SN_{\mathcal{HD}}$ (0.868) by 0.5%. This slight edge reflects $RA_{\mathcal{HD}}$'s strength in recall, which is particularly beneficial for densely connected, group centric datasets.

- **Precision Analysis**
  Precision evaluates the accuracy of positive link predictions while minimizing false positives. For the NDC-classes,email-Eu and hyperedges-contact-high-school datasets which are in Table.4, $RA_{\mathcal{HD}}$ achieves the highest precision of 0.513, 0.616 and 0.512, which is 11%, 8.6%, and 1.6% higher than $SN_{\mathcal{HD}}$. Resource allocation methods prioritize accurate identification of true links, making them particularly effective in hypergraphs with well-defined and also emphasizes on direct node participation within hyperedges enables resource allocation methods to excel in diverse and large-scale

Table 5 Performance of centrality based LP in graph Vs hypergraph modes. $\mathcal{CC}$ and $\mathcal{HCC}$ represent Clustering Coefficient in the graph and hypergraph models respectively. *SN* and *RA* signify the Link Prediction (LP) measures of Shared Neighbor and Resource Allocation

Evaluation Metrics	AUPR				F1-score				Precision			
LP Measures	$SN_{\mathcal{CC}}$	$SN_{\mathcal{HCC}}$	$RA_{\mathcal{CC}}$	$RA_{\mathcal{HCC}}$	$SN_{\mathcal{CC}}$	$SN_{\mathcal{HCC}}$	$RA_{\mathcal{CC}}$	$RA_{\mathcal{HCC}}$	$SN_{\mathcal{CC}}$	$SN_{\mathcal{HCC}}$	$RA_{\mathcal{CC}}$	$RA_{\mathcal{HCC}}$
NDC-classes	0.022	**0.033**	0.013	0.019	0.211	**0.898**	0.198	0.812	0.198	**0.496**	0.208	0.489
email-Eu	0.062	0.059	0.055	**0.083**	0.292	**0.891**	0.258	0.792	0.257	0.501	0.255	**0.589**
c-e-g-q	0.061	0.322	0.083	**0.339**	0.245	0.702	0.278	**0.789**	0.221	**0.512**	0.133	0.496
h-c-h-s	0.043	0.086	0.072	**0.129**	0.161	0.886	0.271	**0.893**	0.131	0.496	0.141	**0.507**

communication networks and also Hyperdegree centrality's focus on immediate connectivity ensures highly accurate predictions for socially clustered interactions. For the cat-edge-geometry-questions dataset, $SN_{\mathcal{HD}}$ achieves the highest precision of 0.519, which is 3.4% higher than $RA_{\mathcal{HD}}$. This result indicates that degree-based measures perform better in geometric datasets, where simpler connectivity patterns dominate. The analysis demonstrates that hypergraph based measures, particularly $SN_{\mathcal{HD}}$ and $RA_{\mathcal{HD}}$, consistently outperform traditional graph-based centralities across all datasets and metrics. Hyperdegree centrality $SN_{\mathcal{HD}}$ excels in AUPR and F1-score due to its ability to capture dense connectivity and multi-node interactions within hyperedges. In contrast, resource allocation-based hyperdegree $RA_{\mathcal{HD}}$ achieves superior precision by effectively minimizing false positives. These findings highlight the advantages of hypergraph based methods in modeling complex interaction patterns and predicting future connections in datasets characterized by multi-node relationships.

Evaluation of the remaining centrality measures, including Clustering Coefficient ($\mathcal{CC}$) and HyperClustering Coefficient ($\mathcal{HCC}$) which is in Table.5, Betweenness ($\mathcal{B}$) and HyperBetweenness ($\mathcal{HB}$) in Table.6, as well as Closeness ($\mathcal{CL}$) and HyperCloseness ($\mathcal{HCL}$) in Table.7, reveals consistent trends in the data sets for AUPR, F1 score and Precision. For the Clustering Coefficient ($\mathcal{CC}$) and HyperClustering Coefficient ($\mathcal{HCC}$) in Table.5, $CN_{\mathcal{HCC}}$ outperforms its traditional counterpart $CN_{\mathcal{CC}}$ in most datasets. In the NDC-classes dataset, $CN_{\mathcal{HCC}}$ achieves the highest AUPR of 0.024, which is 9% higher than $CN_{\mathcal{CC}}$, while its F1-score is 0.898, exceeding $RA_{\mathcal{HCC}}$ by 14.2%. This underscores the importance of hyperclustering in capturing dense local structures within hyperedges. Similarly, in the email-Eu dataset, $CN_{\mathcal{HCC}}$ achieves the best AUPR of 0.121, 98% higher than $CN_{\mathcal{CC}}$, while its F1-score reaches 0.891, further emphasizing the effectiveness of hyperclustering in multi-node interactions. However, for Precision, $RA_{\mathcal{HCC}}$ performs slightly better in some datasets, such as email-Eu, where it achieves a Precision of 0.525, indicating that resource allocation is more effective at minimizing false positives in these settings. For datasets like cat-edge-geometry-questions and hyperedges-contact-high-school, $RA_{\mathcal{HCC}}$ dominates in AUPR, achieving 0.313 and 0.086, respectively, highlighting the role of resource allocation in scenarios involving cohesive group structures.

When considering Betweenness ($\mathcal{B}$) and HyperBetweenness ($\mathcal{HB}$) from Table.6, $SN_{\mathcal{HB}}$ consistently outperforms $SN_{\mathcal{HB}}$ in AUPR across all datasets. In the NDC-classes dataset, $SN_{\mathcal{HB}}$ achieves an AUPR of 0.022, which is

Table 6 Performance of centrality based LP in graph Vs hypergraph modes. $\mathcal{B}$ and $\mathcal{HB}$ represent Betweenness in the graph and hypergraph models respectively. SN and RA signify the Link Prediction (LP) measures of Shared Neighbor and Resource Allocation

Evaluation Metrics	AUPR				F1-score				Precision			
LP Measures	$SN_{\mathcal{B}}$	$SN_{\mathcal{HB}}$	$RA_{\mathcal{B}}$	$RA_{\mathcal{HB}}$	$SN_{\mathcal{B}}$	$SN_{\mathcal{HB}}$	$RA_{\mathcal{B}}$	$RA_{\mathcal{HB}}$	$SN_{\mathcal{B}}$	$SN_{\mathcal{HB}}$	$RA_{\mathcal{B}}$	$RA_{\mathcal{HB}}$
NDC-classes	0.013	**0.022**	0.011	0.016	0.196	**0.837**	0.177	0.779	0.114	**0.489**	0.186	0.471
email-Eu	0.063	**0.092**	0.049	**0.075**	0.182	**0.829**	0.156	0.766	0.143	0.513	0.249	**0.548**
c-e-g-q	0.016	0.233	0.049	**0.289**	0.189	0.622	0.208	**0.705**	0.213	**0.483**	0.112	0.478
h-c-h-s	0.071	0.042	0.031	**0.057**	0.152	0.841	0.119	**0.867**	0.127	0.471	0.133	**0.493**

69.2% higher than $SN_{\mathcal{B}}$. Similarly, in the email-Eu dataset, $SN_{\mathcal{HB}}$ scores 0.092, 46% higher than $SN_{\mathcal{B}}$. These results indicate that hyperbetweenness effectively captures multi-node bridging roles within hypergraphs. While $SN_{\mathcal{HB}}$ often excels in F1-score due to its ability to balance precision and recall, $RA_{\mathcal{HB}}$ demonstrates its strength in Precision, particularly in datasets like cat-edge-geometry-questions, where it achieves a Precision of 0.483, 2.5% higher than $SN_{\mathcal{HB}}$. In the hyperedges-contact-high-school dataset, $RA_{\mathcal{HB}}$ achieves the highest F1-score of 0.831, surpassing $SN_{\mathcal{HB}}$ by 6.5%, highlighting its ability to model group-based social interactions effectively. For Closeness ($\mathcal{CL}$) and HyperCloseness ($\mathcal{HCL}$) from Table.7, $SN_{\mathcal{HCL}}$ consistently outperforms $CN_{\mathcal{CL}}$ in AUPR and F1-score across all datasets. In the NDC-classes dataset, $SN_{\mathcal{HCL}}$ achieves an AUPR of 0.027, 28.6% higher than $SN_{\mathcal{CL}}$, and an F1-score of 0.798, surpassing $RA_{\mathcal{HCL}}$ by 4.6%. This trend is also evident in the email-Eu dataset, where $SN_{\mathcal{HCL}}$ records an AUPR of 0.97, 70.3% higher than $SN_{\mathcal{CL}}$, and an F1-score of 0.812. In contrast, $RA_{\mathcal{HCL}}$ demonstrates its strength in Precision, particularly in the email-Eu dataset, where it achieves 0.528, 6% higher than $SN_{\mathcal{HCL}}$. In the cat-edge-geometry-questions dataset, $RA_{\mathcal{HCL}}$ achieves the highest AUPR of 0.312, surpassing $SN_{\mathcal{HCL}}$ by 8.7%, emphasizing the effectiveness of resource allocation in modeling localized connectivity. However, for F1-score and Precision, $SN_{\mathcal{HCL}}$ remains competitive, particularly in datasets like hyperedges-contact-high-school, where it achieves a Precision of 0.497, slightly outperforming $RA_{\mathcal{HCL}}$.

Overall, these results demonstrate the advantages of hypergraph based measures over their graph based counterparts. The hypergraph specific adaptations $\mathcal{HCC}$, $\mathcal{HB}$, and $\mathcal{HCL}$ consistently outperform traditional centralities $\mathcal{CC}$, $\mathcal{B}$, and $\mathcal{CL}$ across AUPR, F1-score, and Precision. This highlights the ability of hypergraph based methods to capture complex, multi-node interactions within hyperedges, thereby providing a more accurate prediction of future links in diverse datasets. Additionally, resource allocation based hypercentralities $RA_{\mathcal{HCC}}$, $RA_{\mathcal{HB}}$ and $RA_{\mathcal{HCL}}$ often perform well in Precision due to their emphasis on minimizing false positives, making them suitable for datasets where accurate link prediction is critical.

In addition to the detailed tabular analysis presented earlier, the corresponding visual representations provided in Fig. 3a, b, Fig. 4a, b collectively offer a clear and intuitive understanding of the performance trends across all datasets and evaluation metrics. These figures present a pictorial summary of the same results explained in the tables, enabling a direct comparison between graph based and hypergraph based approaches across multiple centrality perspectives, including Degree, Closeness, Betweenness, and Clustering Coefficient. The visual patterns consistently

Table 7 Performance of centrality based LP in graph Vs hypergraph modes. $\mathcal{CL}$ and $\mathcal{HCL}$ represent Closeness in the graph and hypergraph models respectively. SN and RA signify the Link Prediction (LP) measures of Shared Neighbor and Resource Allocation

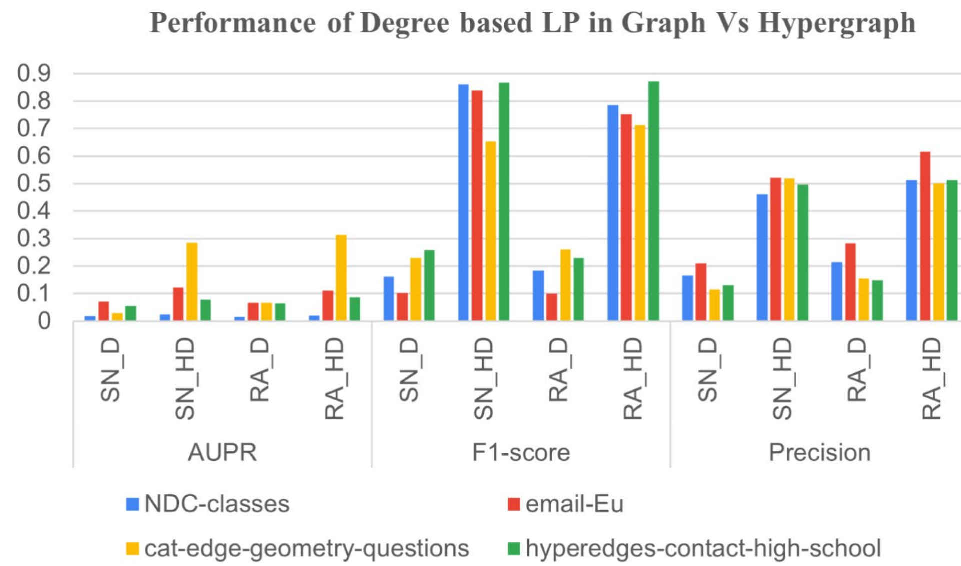
Evaluation Metrics	AUPR				F1-score				Precision			
LP Measures	$SN_{\mathcal{CL}}$	$SN_{\mathcal{HCL}}$	$RA_{\mathcal{CL}}$	$RA_{\mathcal{HCL}}$	$SN_{\mathcal{CL}}$	$SN_{\mathcal{HCL}}$	$RA_{\mathcal{CL}}$	$RA_{\mathcal{HCL}}$	$SN_{\mathcal{CL}}$	$SN_{\mathcal{HCL}}$	$RA_{\mathcal{CL}}$	$RA_{\mathcal{HCL}}$
NDC-classes	0.021	**0.027**	0.015	0.018	0.123	**0.798**	0.152	0.752	0.189	**0.492**	0.189	0.467
email-Eu	0.066	**0.089**	0.051	0.069	0.202	**0.812**	0.148	0.741	0.153	0.518	0.212	**0.577**
c-e-g-q	0.027	0.111	0.052	**0.243**	0.192	0.512	0.202	**0.688**	0.215	**0.498**	0.121	0.476
h-c-h-s	0.063	**0.051**	0.027	0.039	0.156	0.838	0.127	**0.841**	0.029	0.462	0.118	**0.477**

reinforce the quantitative findings reported earlier: hypergraph based variants of Shared Neighbors and Resource Allocation exhibit superior performance across AUPR, F1-score, and Precision, regardless of the underlying centrality framework. By bringing together multiple centrality driven analyses into unified graphical formats, the figures strengthen the interpretation of performance differences and provide an immediate comparative view of how effectively hypergraph modeling captures higher-order interactions, multi-node participation, and complex structural dependencies within real-world datasets.

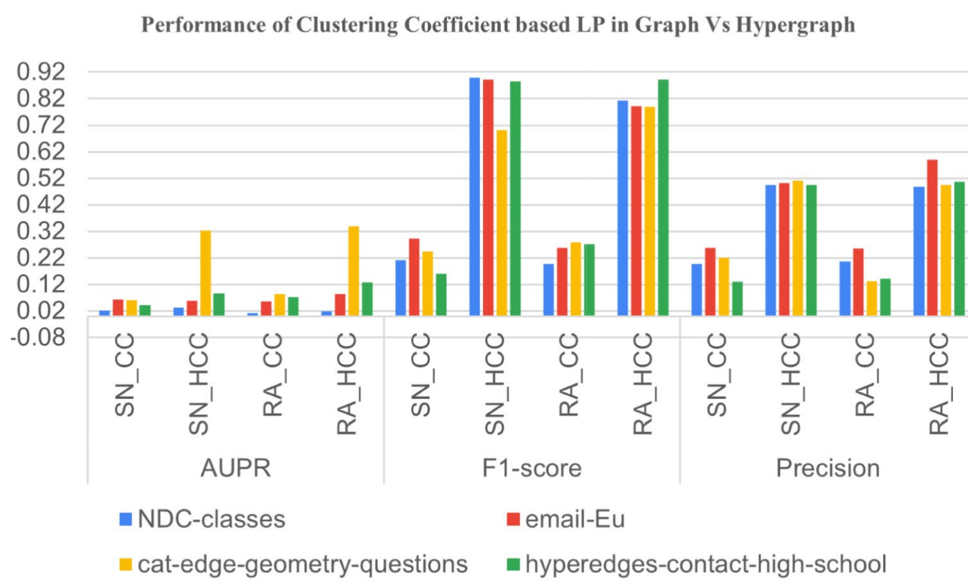
Discussion

In the discussion section, we can conduct a comparative analysis of the highest performing Shared Neighbor and Resource Allocation based hypercentrality measures across each dataset, as follows: In terms of AUPR, $SN_{\mathcal{HCC}}$ consistently outperforms all other measures, including $RA_{\mathcal{HC}}$. This superior performance of $SN_{\mathcal{HCC}}$ can be attributed to its effectiveness in capturing dense local structures within hyperedges, a crucial factor in datasets where clustering dynamics are predominant. In contrast, Resource Allocation based measures, such as $RA_{\mathcal{HCC}}$, emphasize the distribution of resources across connections, which may not encapsulate clustering to the same extent as the Shared Neighbor-based approach. Comparing $SN_{\mathcal{HCC}}$ to the lowest performing measure, $RA_{\mathcal{HCL}}$, in this dataset demonstrates an approximate improvement of 97%, underscoring the robustness of $SN_{\mathcal{HCC}}$ in predicting future links based on clustering properties. For the email-Eu dataset, $SN_{\mathcal{HD}}$ achieves the highest AUPR among the proposed measures, highlighting the effectiveness of hyperdegree centrality in settings characterized by high communication volumes across numerous nodes. The focus on hyperdegree enables $SN_{\mathcal{HD}}$ to capture the frequency and intensity of direct connections, a critical feature in email communication networks. Although Resource Allocation-based measures, such as $RA_{\mathcal{HD}}$, also utilize hyperdegree, the Shared Neighbor approach provides a better representation of direct connectivity patterns. Here, $SN_{\mathcal{HD}}$ performs approximately 75.3% better than least $RA_{\mathcal{HCL}}$ measure, emphasizing the advantage of focusing on shared neighbors in networks with frequent interactions. In the cat-edge-geometry dataset, $RA_{\mathcal{HCC}}$ demonstrates the highest performance, surpassing $SN_{\mathcal{HCC}}$. This dataset likely consists of geometrically similar nodes that form cohesive groups, which aligns well with the Resource Allocation approach that distributes resources across clusters. The hyperclusteringcoefficient within the Resource Allocation framework is particularly effective at capturing these localized group interactions, as it reflects the

Fig. 3 Performance comparison of Centrality based Link Prediction in Graphs and Hypergraphs using AUPR, F1-score, and Precision



(a) Performance comparison of Degree based Shared Neighbors and Resource Allocation methods in Graphs and Hypergraphs using AUPR, F1-score, and Precision

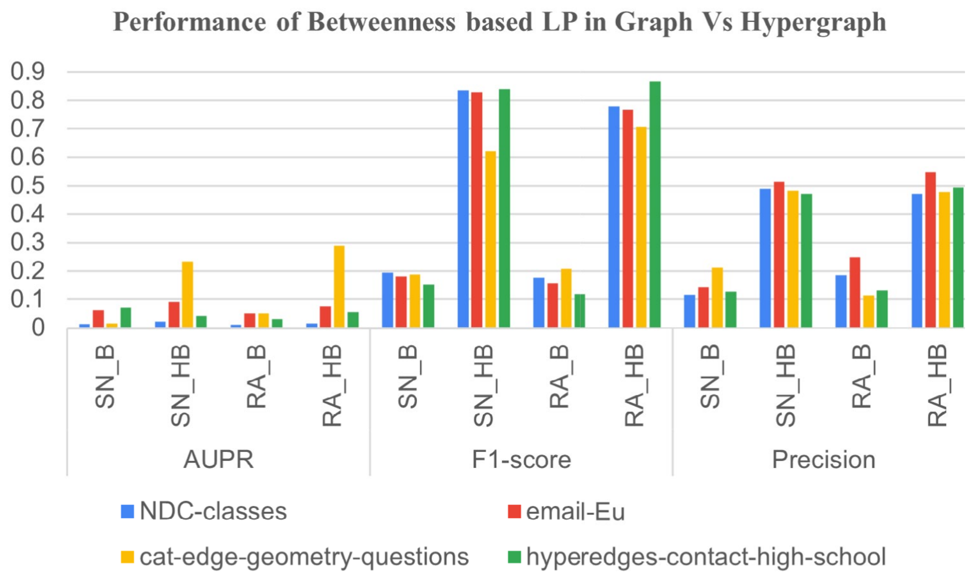


(b) Performance comparison of Clustering Coefficient based Shared Neighbors and Resource Allocation methods in Graphs and Hypergraphs using AUPR, F1-score, and Precision

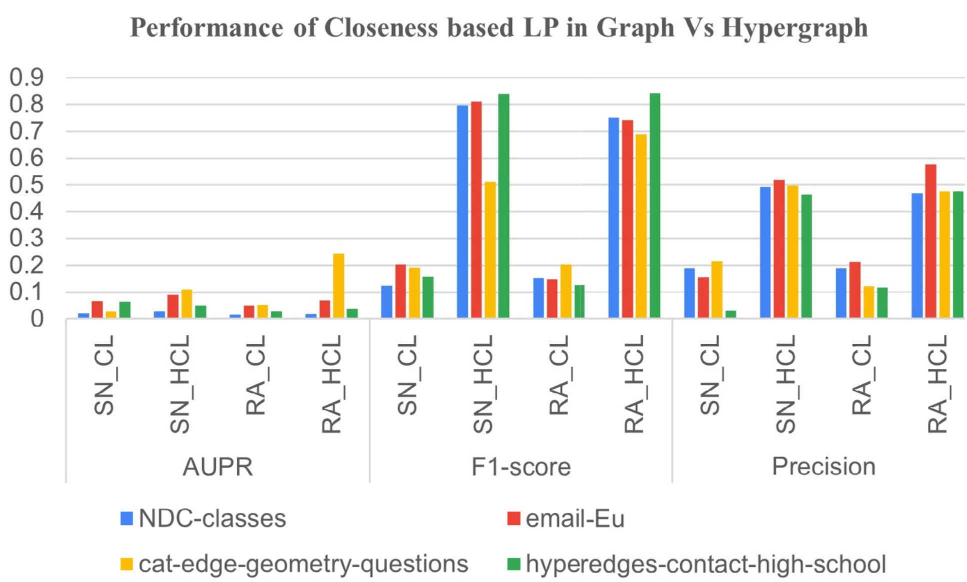
resource concentration within cohesive clusters. Similarly, $RA_{\mathcal{H}CC}$ achieves the highest performance in the contact-high-school dataset. This dataset likely consists of tightly knit communities where frequent group-based interactions occur, such as student groups or classes. The Resource Allocation approach with hyperclustering ($RA_{\mathcal{H}CC}$) effectively captures these frequent interactions by distributing link prediction weight across clusters, aligning closely with group-based social dynamics. Compared to $SN_{\mathcal{H}CC}$, $RA_{\mathcal{H}CC}$ shows an superiority, highlighting the importance of resource-based distribution in hypergraphs with strong community structures.

For the F1-score analysis between Shared Neighbor-based and Resource Allocation based hypercentralities in hypergraphs, the following insights emerge based on the top performing and lowest performing measures across the datasets. In the NDC-classes and email-Eu datasets, $SN_{\mathcal{H}CC}$ achieves the highest F1-scores. This success can be attributed to the Shared Neighbor approach's strength in datasets where clustering and group dynamics are central.

Fig. 4 Performance comparison of Centrality based Link Prediction in Graphs and Hypergraphs using AUPR, F1-score, and Precision



(a) Performance comparison of Betweenness based Shared Neighbors and Resource Allocation methods in Graphs and Hypergraphs using AUPR, F1-score, and Precision



(b) Performance comparison of Closeness based Shared Neighbors and Resource Allocation methods in Graphs and Hypergraphs using AUPR, F1-score, and Precision

The $SN_{\mathcal{HCC}}$ measure leverages clustering properties within hyperedges, allowing it to effectively capture dense, cohesive structures characteristic of NDC-classes and email-Eu. This focus on clustering aligns with the structure of hypergraphs in these datasets, where connections are typically concentrated within distinct groups. When $SN_{\mathcal{HCC}}$ is compared to the least-performing Resource Allocation-based measure, $RA_{\mathcal{HCL}}$, a notable performance difference is observed. In the NDC-classes dataset, $SN_{\mathcal{HCC}}$ outperforms $RA_{\mathcal{HCL}}$ by approximately 12.5% in F1-score. Similarly, in the email-Eu dataset, $SN_{\mathcal{HCC}}$ shows an improvement of 9.4% over $RA_{\mathcal{HCL}}$. These differences indicate that Shared Neighbor based measures, particularly those emphasizing clustering, are more effective at capturing the connection patterns in these datasets than Resource Allocation based measures, which focus on resource distribution and may not as strongly emphasize local clustering. In contrast, within the cat-edge-geometry and contact-high-school datasets, $RA_{\mathcal{HCC}}$ achieves the highest F1-scores, surpassing Shared Neighbor-based measures. The effectiveness of Resource Allocation in these datasets can be linked to its approach of distributing prediction weight across clusters, effectively capturing relational dynamics in datasets where nodes frequently interact within tightly bound communities. Resource Allocation-based hypercentralities model group-based interactions particularly well, especially in contexts where the network structure highlights multi-node participation in hyperedges. When $RA_{\mathcal{HCC}}$ is compared to the lowest performing $SN_{\mathcal{HCL}}$, a substantial performance difference is evident. In the cat-edge-geometry dataset, $RA_{\mathcal{HCC}}$ outperforms $SN_{\mathcal{HCL}}$ by approximately 27.6%. Similarly, in the contact-high-school dataset, $RA_{\mathcal{HCC}}$ demonstrates an advantage of about 29.5% over $SN_{\mathcal{HCL}}$. This significant difference underscores the importance of Resource Allocation in accurately modeling resource distribution within densely interconnected clusters, a crucial feature in community-oriented datasets like cat-edge-geometry and contact-high-school.

Although hypergraph centrality distributions may exhibit heterogeneity or skewness, the empirical results across all datasets demonstrate that the global average hypercentrality serves as an effective and reliable threshold within the CLPH framework. The average provides a network wide structural baseline that is entirely parameter free and naturally adapts to each dataset, avoiding the added complexity of percentile or region specific thresholds. Importantly, the superior performance of the proposed measures indicates that this threshold successfully suppresses low influence neighbours without excluding structurally significant ones, thereby maintaining the predictive stability observed throughout our evaluation.

Conclusion and Future Work

In this paper, we proposed a link prediction framework that employs hypercentrality measures to address the intricacies of multi-node interactions within hypergraphs. By adapting traditional centrality metrics such as degree, clustering coefficient, betweenness, and closeness to the hypergraph context, we developed a novel link prediction model that leverages the rich structural information of hypergraphs. This approach enhances traditional link prediction methods by incorporating node importance through hypercentrality scores. Our empirical analysis, conducted across multiple real-world datasets, demonstrates that hypercentrality-based models, particularly those utilizing hyperdegree and hyperclustering coefficients, consistently surpass existing link prediction based centrality approaches in terms of AUPR, F1-score and Precision. These findings suggest that link prediction based hypercentrality measures offer a more precise and nuanced approach for predicting link formation in hypergraphs, especially in graphs characterized by dense connectivity and pronounced clustering.

This study focuses exclusively on undirected, unweighted, and static hypergraphs, and the reported results may therefore not directly generalize to dynamic or multi-modal systems. Extending the CLPH framework to incorporate temporal evolution, weighted interactions, and multi-layer structural information constitutes an important direction for future work, enabling its applicability to dynamic financial networks, evolving social platforms, and heterogeneous real-world hypergraph data. Future research will explore hybrid models in which centrality weighted shared neighbour features derived from CLPH are incorporated into supervised learning pipelines or hypergraph neural architectures. Such integration may enhance predictive performance, particularly for large-scale, dynamic, or richly annotated hypergraph datasets. For future research, we intend to expand our exploration beyond local based to global based measures. Additionally, rather than considering all non-adjacent node pairs, we plan to investigate the impact of limiting the prediction scope to node pairs within specific hop distances, such as 2, 3, and beyond. Another key direction is to extend our current model—which focuses on non-adjacent node pairs of size 2 to include interactions among groups of size 3, 4, and larger, thereby capturing more complex group dynamics. These extensions will enable a more comprehensive understanding of link prediction in hypergraphs, further enhancing the model's applicability to real-world scenarios.

Author Contributions The concept was conceived by Jaya Lakshmi, who also handled the documentation. Y.V.Nandini was responsible for the design and implementation, while Murali Krishna Enduri contributed by defining the notation and analyzing the results.

Funding Not applicable.

Data Availability Not applicable.

Declaration

Conflict of Interest The authors do not have any conflict of interest.

Research Involving Human and /or Animals Not applicable.

Informed Consent Not applicable.

References

1. Albert R, Barabási A-L. Statistical mechanics of complex networks. Rev Mod Phys. 2002;74(1):47.
2. Vazquez A. Complex hypergraphs. Phys Rev E. 2023;107(2):024316.
3. Liben-Nowell D, Kleinberg J. The link prediction problem for social networks. In: Proceedings of the Twelfth International Conference on Information and Knowledge Management, 2003; pp. 556–559.
4. Wu H, Yan Y, Ng MK-P. Hypergraph collaborative network on vertices and hyperedges. IEEE Trans Pattern Anal Mach Intell. 2022;45(3):3245–58.
5. Feng S, Heath E, Jefferson B, Joslyn C, Kvinge H, Mitchell HD, et al. Hypergraph models of biological networks to identify genes critical to pathogenic viral response. BMC Bioinform. 2021;22(1):287.
6. Chen C, Liu Y-Y. A survey on hyperlink prediction. IEEE Transactions on Neural Networks and Learning Systems 2023
7. Saxena A, Iyengar S. Centrality measures in complex networks: a survey. 2020. arXiv preprint arXiv:2011.07190.
8. Nandini Y, Lakshmi TJ, Enduri MK, Sharma H. Link prediction in complex networks using average centrality-based similarity score. Entropy. 2024;26(6):433.
9. Sahoo KS, Tripathy BK, Naik K, Ramasubbareddy S, Balusamy B, Khari M, et al. An evolutionary svm model for ddos attack detection in software defined networks. IEEE Access. 2020;8:132502–13.
10. Vimal S, Khari M, Crespo RG, Kalaivani L, Dey N, Kaliappan M. Energy enhancement using multiobjective ant colony optimization with double q learning algorithm for iot based cognitive radio networks. Comput Commun. 2020;154:481–90.
11. Rajagopal A, Ramachandran A, Shankar K, Khari M, Jha S, Lee Y, et al. Fine-tuned residual network-based features with latent variable support vector machine-based optimal scene classification model for unmanned aerial vehicles. IEEE Access. 2020;8:118396–404.
12. Rajagopal A, Ramachandran A, Shankar K, Khari M, Jha S, Joshi GP. Optimal routing strategy based on extreme learning machine with beetle antennae search algorithm for low earth orbit satellite communication networks. Int J Satell Commun Netw. 2021;39(3):305–17.
13. Addula SR, Kumar MK, Kumar D, Ananthan B. et al. Dynamic load balancing in cloud computing using hybrid kookaburra-pelican optimization algorithms. In: 2024 International Conference on Augmented Reality, Intelligent Systems, and Industrial Automation (ARIIA), 2024; pp. 1–7. IEEE
14. Estrada E, Rodriguez-Velazquez JA. Complex networks as hypergraphs. 2005. arXiv preprint physics/0505137.
15. Kumar T, Darwin K, Parthasarathy S, Ravindran B. Hpra: Hyperedge prediction using resource allocation. In: Proceedings of the 12th ACM Conference on Web Science, 2020; pp. 135–143.
16. Nandini Y, Lakshmi TJ, Enduri MK, Sharma H, Ahmad MW. Extending graph-based lp techniques for enhanced insights into complex hypergraph networks. IEEE Access. 2024.
17. Freeman LC, et al. Centrality in social networks: Conceptual clarification. Social network: critical concepts in sociology. Londres: Routledge. 2002;1:238–63.
18. Serrano MÁ, Boguna M. Clustering in complex networks. i. General formalism. Phys Rev E-Stat Nonlinear Soft Matter Phys. 2006;74(5):056114.
19. Costantini G, Perugini M. Generalization of clustering coefficients to signed correlation networks. PLoS ONE. 2014;9(2):88669.
20. Freeman L. A set of measures of centrality based on betweenness. Sociometry. 1977.
21. Newman M. Networks. Oxford University Press; 2018.
22. Roy S, Ravindran B. Measuring network centrality using hypergraphs. In: Proceedings of the 2nd ACM IKDD Conference on Data Sciences, 2015; pp. 59–68.
23. Li D, Xu Z, Li S, Sun X.: Link prediction in social networks based on hypergraph. In: Proceedings of the 22nd International Conference on World Wide Web, 2013; pp. 41–42.
24. Tuğal İ, Pala Z. Centrality with entropy in hypergraphs based on similarity measures. Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi. 2023;14(3):407–19.
25. Aksoy SG, Joslyn C, Marrero CO, Praggastis B, Purvine E. Hypernetwork science via high-order hypergraph walks. EPJ Data Sci. 2020;9(1):16.
26. Benson AR, Abebe R, Schaub MT, Jadbabaie A, Kleinberg J. Simplicial closure and higher-order link prediction. Proc Natl Acad Sci. 2018. https://doi.org/10.1073/pnas.1800683115.
27. Mastrandrea R, Fournet J, Barrat A. Contact patterns in a high school: A comparison between data collected using wearable sensors, contact diaries and friendship surveys. PLoS ONE. 2015;10(9):0136497. https://doi.org/10.1371/journal.pone.0136497.
28. Boyd K, Eng KH, Page CD. Area under the precision-recall curve: point estimates and confidence intervals. In: Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III 13, 2013; pp. 451–466. Springer
29. Yang Y, Lichtenwalter RN, Chawla NV. Evaluating link prediction methods. Knowl Inf Syst. 2015;45:751–82.
30. Jiao X, Wan S, Liu Q, Bi Y, Lee Y-L, Xu E, et al. Comparing discriminating abilities of evaluation metrics in link prediction. J Phys Complex. 2024;5(2):025014.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.