Beyond Pairwise Links: Hypergraph Modeling for Scientific Trend Forecasting

Nithyasree Kusakula[1], Navyamsh Gangavaram[1], Liz Torres[1], Russell J. Funk[2], and Mehmet Emin Aktas[3]

Georgia State University, Atlanta, GA, USA
{nkusakula1, ngangavaram1, lgarcia46}@student.gsu.edu
University of Minnesota, Minneapolis, MN, USA
rfunk@umn.edu
Kennesaw State University, Marietta, GA, USA
maktas1@kennesaw.edu

Abstract. Scientific research is expanding rapidly, producing vast volumes of scholarly literature across diverse disciplines. However, identifying emerging trends and uncovering interdisciplinary intersections remains a persistent challenge due to the fragmented and nonlinear nature of modern knowledge production. Traditional bibliometric methods such as co-citation analysis and keyword co-occurrence rely on pairwise relationships and often fail to capture the higher-order associations that drive innovation. In this study, we present a novel hypergraph-based framework for forecasting scientific research trends. In our approach, nodes represent research concepts, and hyperedges correspond to publications that link multiple concepts. By framing the task as a hyperedge link prediction problem, we uncover latent conceptual groupings that may signal future research directions. We then develop hypergraph neural network models, HLP, HyperGCN, and HyperSAGE, alongside traditional graph-based models, using a real-world dataset of scientific concepts used in papers from the field of Ethnic Studies. Our results show that hypergraph models consistently outperform their graph-based counterparts in accuracy and predictive power.

Keywords: Hyperedge Link Prediction, Scientific Trend Forecasting, Hypergraph Neural Networks, Research Mapping

1 Introduction

Understanding and predicting scientific trends is a crucial task with far-reaching implications for researchers, funding agencies, and policymakers (10; 3; 28). Accurately forecasting emerging areas of inquiry can guide strategic investments, foster timely interdisciplinary collaborations, and accelerate scientific discovery. The rapid expansion of scientific literature has made it increasingly challenging to detect emerging research trends and anticipate interdisciplinary developments. Traditional bibliometric techniques, such as co-authorship analysis, co-citation

networks, and keyword co-occurrence graphs, primarily rely on pairwise relationships and often fall short in capturing the complex, higher-order interactions that underpin knowledge creation and dissemination across domains (4; 26).

On the other hand, hypergraphs offer a sophisticated framework for modeling complex, higher-order interactions across various domains such as biology, chemistry, finance, and social sciences (6; 30; 24; 25). Unlike traditional networks that focus on pairwise connections, hypergraphs represent entities as nodes and group interactions as hyperedges, enabling a more nuanced capture of real-world interactions.

In this paper, we propose novel hypergraph-based models to predict emerging scientific trends by capturing the higher-order relationships that are not well represented by traditional network approaches. We first represent research articles as a hypergraph where keywords are nodes and articles are hyperedges. In contrast to simple graphs that represent only binary interactions, this hypergraph models relationships among arbitrary-sized groups of keywords, thereby offering a richer and more expressive framework for representing scholarly networks (5; 33).

As the next step, we integrate advanced hypergraph embedding techniques with deep learning models to capture both structural and contextual patterns within the hypergraph. We consider the scientific research trend prediction problem as hyperedge link prediction, where each predicted hyperedge represents a new research article. This model supports a more accurate understanding of the evolving landscape of scientific innovation (7; 9). While hyperedge prediction remains relatively underexplored compared to traditional link prediction in graphs (31; 16), its application in modeling scholarly ecosystems represents a novel and impactful direction for both graph theory and science-of-science analytics.

To understand the effectiveness and efficiency of the proposed hypergraph models, we compare these models with their counterpart graph models. We first create a graph where nodes again represent keywords, and we add an edge between two nodes when the corresponding keywords happen to be in the same research article. We then build graph embedding and deep learning models for link prediction in the same way as the hypergraph. Our experimental results on the Ethnic Studies dataset show that our hypergraph models are more effective and efficient in trend forecasting and knowledge discovery than baseline graph models. Our models extend beyond academia. They have practical implications for industry stakeholders in academic publishing, educational services, and research funding, and enable data-driven decisions that prioritize emerging fields with high growth potential.

The paper is formatted as follows. In Section 2, we discuss the preliminary concepts for graphs and hypergraphs, and present related work. In Section 3, we present our methodology on hypergraph representation of research articles and prediction models. In Section 4, we explain our evaluation method, the baseline models, and present our results on a dataset on Ethnic Studies. Our final remarks with future work directions are found in Section 5.

2 Background

2.1 Preliminaries

Graphs are structured data representing relationships between objects (1; 8). They are formed by a set of *vertices* (also called nodes) and a set of *edges* that are connections between pairs of vertices. In a formal definition, a network G is a pair of sets $G = (V, E)$ where V is the set of vertices and $E \subset V \times V$ is the set of edges of the network. If there is a score for the relationship between vertices that could represent the strength of interaction, we can represent this type of relationship or interaction by a *weighted network*. In a weighted network, a weight function $w : E \to \mathbb{R}$ is defined to assign a weight for each edge.

Let G be a weighted undirected graph with the vertex set V and a weight function $w : V \times V \to \mathbb{R}^{\geq 0}$. The *adjacency matrix* A of G is defined as the $n \times n$ matrix with $A(i, j) = w(v_i, v_j)$ for $i, j \in \{1, ..., n\}$ with n being the number of vertices of G. Furthermore, let D be the $n \times n$ diagonal matrix with $D(i, i) = \sum_j w(i, j)$, i.e., the weighted degree of the vertex $i \leq n$. We can define the graph Laplacian L as $L = D - A$ where D is the weighted degree matrix and A is the weighted adjacency matrix. The graph Laplacian has been thoroughly studied and applied in network science for years (19). The spectrum of Laplacian is related to many graph features such as connected components, spanning trees, centralities, and diffusion (20).

On the other hand, a *hypergraph* H denoted by $H = (V, E = (e_i)_{i \in I})$ on the finite vertex set V is a family $(e_i)_{i \in I}$ (I is a finite set of indexes) of subsets of V called *hyperedges*. Hyperedges can be of different sizes, possibly ranging from one vertex $\{v\} \subseteq V$ to the entire vertex set V.

2.2 Related Work

The task of forecasting emerging trends in scientific research has received increasing attention in recent years, driven by the explosive growth of scholarly literature and the need for more proactive scientific foresight. A wide range of approaches have been proposed, from citation-based heuristics to deep learning and graph-based models that capture the complex structure of scientific knowledge.

The authors in (18) introduced *SEMNET*, a semantic network of over 750,000 physics papers in which concepts are represented as nodes and their co-occurrences as edges. By training a deep neural network on the temporal evolution of this network, they successfully forecasted emerging developments in quantum physics, achieving an AUC of 0.85 in predictive performance.

Zheng et al. proposed a hybrid model that integrates graph convolutional networks (GCN) with temporal convolutional networks (TCNs) to jointly model spatial relationships among topics and their temporal evolution (32). Their approach significantly outperformed traditional baselines in predicting topic "hotness" on dynamic scientific graphs.

Ofer and Linial further advanced this field by combining pre-trained language models (e.g., BERT) with bibliometric indicators such as citation count and recency to predict emerging topics across 125 disciplines (22). Their results emphasize the value of combining semantic understanding with citation-based signals for better generalization.

Complementing these neural approaches, Gipp et al. proposed a co-citation proximity analysis technique for identifying related research documents based on their structural citation context, offering an interpretable alternative for trend detection within scientific corpora (11). Their method demonstrated that semantic relationships inferred from citation positions are useful for clustering and forecasting research evolution.

Several other studies have explored the use of neural or graph-based techniques for bibliometric trend prediction. For example, Zhang et al. developed a graph neural network-based link prediction model, which laid the foundation for many subsequent GCN-based forecasting tasks (31). Similarly, Feng et al. introduced hypergraph neural networks to model higher-order relationships, which align closely with our proposed framework (9).

Together, these efforts underscore the growing interest in modeling scientific progress using neural architectures that can encode both structural and semantic properties of research networks. Building upon these foundations, our work is distinguished by its use of hypergraph neural networks for trend forecasting, an approach that captures multiconcept interactions inherent in research publications, thereby offering more expressive representations and finer-grained predictive insights.

3 Methodology

In this section, we describe how we construct a hypergraph from research articles, and then our hypergraph models to predict scientific trends using hypergraph neural networks.

3.1 Hypergraph Construction

To capture higher-order relationships among research concepts, we represent the scientific corpus as a hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of concepts (e.g., keywords) and $\mathcal{E}$ denotes hyperedges, each corresponding to a research paper. Unlike conventional graphs, hyperedges can connect multiple nodes, allowing the representation of complex semantic groupings common in academic literature (see Figure 1 for an example).

Each paper is treated as a hyperedge connecting the concepts it mentions. We define a sparse incidence matrix $H \in \{0,1\}^{|\mathcal{V}|\times|\mathcal{E}|}$, where $H_{ij} = 1$ indicates that concept $v_i \in \mathcal{V}$ appears in paper $e_j \in \mathcal{E}$. We map each unique concept and paper to numerical indices for efficient matrix construction and processing.

To learn semantic representations of nodes and hyperedges, we employ the Deep Hyperedges Framework (DHE) (29), which extends DeepWalk via two

Fig. 1: Hypergraph representation of the first 10 papers in the dataset (visualized using the HyperNetX package (23)).

random walk strategies: vertex-centric (SaT) and hyperedge-centric (TaS). SaT walks simulate concept cooccurrence by traversing nodes connected through shared hyperedges, with sampling biased toward smaller hyperedges to emphasize specific, less frequent groupings. In contrast, TaS walks treat hyperedges as nodes and model their connections based on shared concepts. Sequences from both views are used to train a Word2Vec model, producing dense embeddings for concepts and hyperedges that encode both structural and semantic contexts.

The prediction of hyperedge links is framed as a supervised binary classification task. Positive samples are actual concept groupings from papers, while negatives are synthetically generated sets of randomly selected keywords that do not co-occur in any paper. All input groups are padded or truncated to a fixed length of 16 for consistency.

3.2 Hypergraph Link Prediction Models

In this section, we present three hypergraph-based models designed to predict the formation of new concept groupings, or hyperedges, within a scientific knowledge

graph. Specifically, we explore (1) the Hyperedge Link Predictor (HLP) model, a neural link predictor that operates on aggregated hyperedge embeddings, (2) the HyperSage model, which performs message passing between nodes and hyperedges, and (3) HyperGCN, a spectral convolutional approach that models each hyperedge as a clique-like structure. Each method offers a different mechanism for learning from the hypergraph structure to identify future research directions.

3.2.1 Hyperedge Link Predictor (HLP) Model: We use the embeddings obtained from DHE as input to a feedforward neural network, referred to as the *Hyperedge Link Predictor* (HLP). For each hyperedge (i.e., concept group), we aggregate its constituent concept embeddings using mean pooling to obtain a fixed-size representation. This vector is passed through two hidden layers with ReLU activation (21) and dropout (27). A final sigmoid layer outputs the probability that the input concept group forms a valid hyperedge.

Training is performed using binary cross-entropy loss with early stopping. Positive samples are derived from the dataset, and negative samples are created by randomly grouping non-co-occurring concepts. All inputs are padded or truncated to length 16. Using high-order relationships and semantically rich embeddings, the model can distinguish plausible interdisciplinary groupings and predict future research trends.

3.2.2 HyperSage Model: To further capture complex higher-order dependencies, we adopt the *HyperSage* architecture (2), which extends GraphSage to hypergraph structures. Each layer alternates between two operations: (1) node-to-hyperedge aggregation using a learnable MLP, and (2) hyperedge-to-node redistribution, enabling message passing through the hypergraph incidence matrix.

Each aggregation step is followed by ReLU activation (21) and dropout (27). The final hyperedge representations are obtained by averaging the embeddings of their constituent nodes, followed by a sigmoid layer to predict whether a concept group constitutes a valid hyperedge. Training uses binary cross-entropy loss and the Adam optimizer (15). This architecture effectively captures both local and global context in the hypergraph structure.

3.2.3 HGCN Model: We also evaluate the *HyperGCN* model (29), which generalizes the convolution of the graph to hypergraphs by modeling each hyperedge as a clique-like substructure. Given the binary incidence matrix H, we compute the normalized Laplacian hypergraph

$$L = D_v^{-1/2} H D_e^{-1} H^\top D_v^{-1/2}$$

where D_v and D_e are diagonal degree matrices for vertices and hyperedges, respectively. This formulation enables message propagation that respects hypergraph topology while ensuring spectral normalization.

HyperGCN processes this structure using stacked graph convolutional layers applied to an initial feature matrix X (e.g., identity or pretrained embeddings). Each layer performs transformation and propagation based on L, with ReLU activation (21) and dropout (27) applied to prevent overfitting. Final node embeddings are aggregated using mean pooling to represent candidate hyperedges, which are scored via a sigmoid output layer. This setup enables generalization from observed groupings to predict emerging topic clusters.

4 Experiments

In this section, we first explain our datasets. Then, we describe the baseline graph link prediction models that we compare. Finally, we present our experimental results on both performance comparison and running times.

4.1 Dataset

The dataset used in this study originates from the Web of Science and covers the field of Ethnic Studies and contains approximately 250,000 records, each representing a research paper. Every paper is associated with a set of research concepts (keywords), which form the basis for constructing both graph and hypergraph representations. Concepts were parsed from paper abstracts using techniques from natural language processing (see (14) for details). Prior to modeling, the dataset was cleaned by removing duplicates, addressing missing values, standardizing concept labels, and excluding papers with fewer than two concepts. All reported statistics reflect the dataset after this cleaning process.

Table 1: Dataset statistics for graph and hypergraph representations.

	$\|V\|$	$\|E\|$	$\langle k \rangle$	k_{max}
Graph	5,362	780,403	291.09	3,248
Hypergraph	5,362	3,849	24.76	596

In the graph representation, edges connect concept pairs that co-occur within the same paper. In the hypergraph representation, each paper is treated as a hyperedge linking all its associated concepts, capturing higher-order interactions. These structures form the foundation of our graph- and hypergraph-based modeling frameworks (see Table 1 for details).

4.2 Baseline Graph Link Prediction Models

We benchmark our hypergraph-based models against standard graph-based link prediction approaches. In this section, we describe how the graph is constructed and the models used for evaluation.

We represent relationships between research concepts using an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each node $v \in \mathcal{V}$ denotes a unique concept and an edge $(v_i, v_j) \in \mathcal{E}$ is formed if the two concepts co-occur in at least one publication. This results in a concept co-occurrence graph created via clique expansion: for each paper, a fully connected subgraph is formed among its associated concepts.

The resulting graph is densely connected, capturing the extent of conceptual overlap across publications. While effective at modeling pairwise relationships, it lacks the ability to represent higher-order interactions—an aspect we address in our hypergraph-based methods.

To learn meaningful concept representations, we apply the *Node2Vec* algorithm (12), which performs biased random walks to explore the graph structure and generate embeddings that encode both homophily and structural roles. These walks balance local and global context through tunable search strategies. The resulting embeddings are trained using the skip-gram model and stored for downstream tasks.

We frame link prediction as a supervised binary classification problem. Positive samples consist of existing edges, while negative samples are sampled from node pairs that are not directly connected. For each edge, we compute its representation by averaging the embeddings of its endpoint nodes, resulting in fixed-size vectors for classification.

This graph-based approach offers a strong and interpretable baseline for learning conceptual associations. However, it remains limited in capturing the rich, multi-concept interactions characteristic of interdisciplinary research, motivating our transition to hypergraph models.

4.2.1 Graph Link Predictor (GLP) Model: To evaluate the graph-based setup, we define the task as binary link prediction: given a pair of concepts, predict whether they co-occur in at least one publication.

To proceed with the classification, we use a lightweight feed-forward neural network (FFNN), trained with binary cross-entropy loss and optimized using Adam. We monitor both training and validation loss during optimization to ensure generalization.

We assess performance on a held-out test set using standard evaluation metrics. We also visualize training dynamics through a loss curve and assess model calibration using an ROC curve. This provides insight into the model's capacity to differentiate valid from invalid concept pairings.

Overall, this baseline highlights the effectiveness of pairwise modeling. In subsequent sections, we compare its performance with hypergraph-based methods to understand the benefits of higher-order structural modeling.

4.2.2 GraphSAGE Model: GraphSAGE is an inductive framework for node representation learning that generates embeddings by sampling and aggregating information from a node's local neighborhood (13). Unlike traditional GCNs, GraphSAGE supports inductive generalization, allowing it to scale to unseen nodes and large graphs by learning aggregation functions instead of storing fixed embeddings.

In this model, each node samples a fixed number of neighbors and applies an aggregation function—such as mean, LSTM, or pooling—to compute its next-layer representation. Embeddings are computed layer by layer, where the representation at layer k is derived from aggregated neighbor features and the node's

own features from layer $k-1$. Formally, for a node v at layer k, the update rule is

$$h_v^{(k)} = \sigma\left(W^{(k)} \cdot \text{AGGREGATE}^{(k)}\left(\left\{h_u^{(k-1)} \mid u \in \mathcal{N}(v)\right\}\right) + b\right) \quad (1)$$

where $\mathcal{N}(v)$ denotes the neighborhood of node v, $W^{(k)}$ and b are learnable parameters, and σ is a non-linear activation function (typically ReLU). The final node embeddings are passed through a sigmoid output layer to predict link existence. Training is done using binary cross-entropy loss and stochastic gradient descent. GraphSAGE is widely adopted for large-scale and dynamic graphs (34; 31).

4.2.3 Graph GCN Model: The Graph Convolutional Network (GCN) is a foundational architecture for learning node embeddings via feature aggregation from local neighborhoods. In our work, we use GCN to model pairwise relationships in the co-occurrence graph. The model consists of multiple graph convolution layers, where each layer refines node embeddings by aggregating neighborhood features followed by a non-linear activation function such as ReLU.

GCN operates on the adjacency matrix A and feature matrix X, using symmetric normalization for stability. The normalized adjacency matrix $\tilde{A}$ is computed as

$$\tilde{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$$

where D is the degree matrix of A. The graph convolution operation for layer l is defined as

$$H^{(l+1)} = \sigma(\tilde{A} H^{(l)} W^{(l)})$$

where $H^{(l)}$ is the node embedding at layer l, $W^{(l)}$ is a trainable weight matrix, and σ is a non-linear activation function. Final node embeddings are passed through a sigmoid scoring layer to classify links between node pairs. The model is trained using binary cross-entropy loss, and dropout is applied to mitigate overfitting (17). GCN provides a strong benchmark by effectively capturing relational patterns through neighborhood aggregation in graph-structured data.

4.3 Results

In this section, we first analyze the performance of the graph and hypergraph models. We further study the training dynamics on the learning curves. Later, we compare the running times of each model. We test and train the graph-based and hypergraph-based models using the same data splits with 85% training, 5% validation, and 10% test ratios.

4.3.1 Performance Analysis: Our experimental results, summarized in Table 2, demonstrate clearly that the hypergraph-based models outperform traditional graph-based approaches in the task of scientific trend prediction. Among the graph-based models, GraphSAGE achieved the highest recall (95.84%) and F1 score (74.51%), outperforming GCN across all metrics. However, these models struggled to balance precision and recall effectively, resulting in moderate overall performance.

Table 2: Performance comparison of the link prediction models. The best results are typed in bold, and the second-best results are underlined.

Model	Accuracy	Precision	Recall	F1 Score
GLP	78.91	81.13	75.34	78.13
GraphSage	67.22	60.95	<u>95.84</u>	74.51
GCN	65.46	59.88	93.66	73.06
HLP	88.16	<u>89.24</u>	86.12	87.65
HyperSage	<u>90.10</u>	88.83	91.74	<u>90.26</u>
HGCN	**98.00**	**98.93**	**96.85**	**97.88**

On the other hand, hypergraph models consistently outperformed their graph-based counterparts across all evaluation metrics. The HGCN model delivered the best overall performance with an accuracy of 98%, precision and recall at 99.93% and 96.85% respectively. HLP and HyperSAGE also achieved strong results, with HyperSage obtaining the second-best scores in most categories, including accuracy (90.10%) and F1 score (90.26%).

The outstanding performance of HGCN can be attributed to its principled use of spectral hypergraph convolution, which models each hyperedge as a clique and performs message passing that reflects the full topological structure of the hypergraph. By leveraging the normalized hypergraph Laplacian, HGCN effectively captures both local and global higher-order interactions among concepts and enables fine-grained generalization from known groupings to novel interdisciplinary clusters. Furthermore, its use of stacked convolutional layers and spectral normalization enhances model stability and expressiveness.

Furthermore, we analyze the AUC-ROC score for each model. AUC-ROC provides a threshold-independent evaluation of model performance. Unlike accuracy, precision, recall, and F1 Score, which depend on a fixed classification threshold, AUC summarizes the model's ability to distinguish between positive and negative samples across all possible thresholds. A higher AUC indicates a better balance between true positive and false positive rates over the full decision space.

As shown in Figure 2, HGCN achieves the highest AUC of 0.99, followed by HyperSage with 0.96, and HLP with 0.9512. These results indicate that hypergraph-based models are more effective than graph-based models (e.g., GraphSAGE: 0.8620; GCN: 0.8499) at capturing the underlying structure of concept co-occurrence and distinguishing meaningful research trend groupings.

We attribute the AUC performance of HGCN to its spectral formulation, which allows it to incorporate global structure and smooth information across concept clusters with high fidelity. The model's use of the normalized hypergraph Laplacian preserves higher-order dependencies and mitigates overfitting, resulting in robust separation between true and false links across thresholds. Similarly, the HyperSAGE model demonstrates a higher AUC than HLP, due to

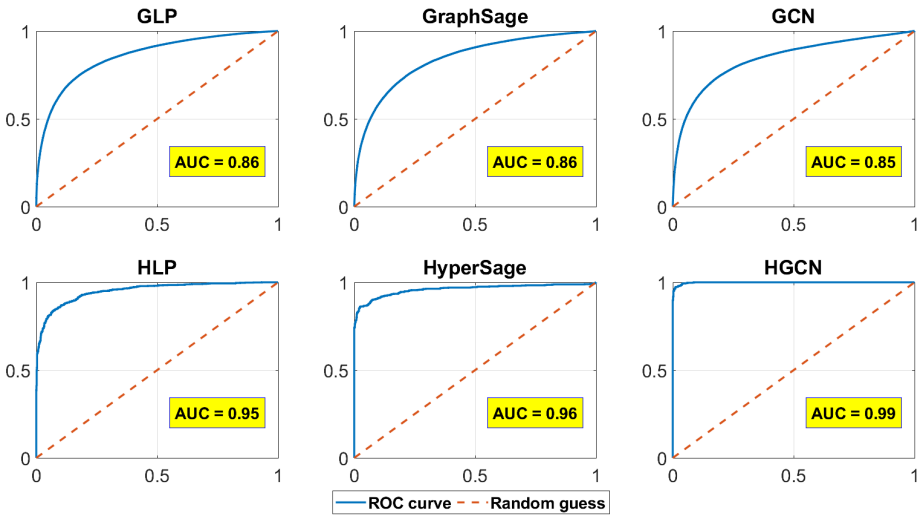

Fig. 2: ROC curves and AUC scores for both graph and hypergraph models.

its alternating node-hyperedge message-passing mechanism that captures both local and global context with greater fidelity.

4.3.2 Training Dynamics Analysis: To evaluate the learning behavior and convergence dynamics of all models, we plotted training and validation losses against epochs (available in Figure 3). Across all models, we observed a consistently smooth decrease in both metrics, indicating stable learning and effective generalization. Notably, the HyperGCN model achieved the lowest final training and validation losses (0.0592 and 0.0499, respectively), demonstrating efficient learning of complex hypergraph structures and strong generalization to unseen data. HyperSAGE and HLP also exhibited steady convergence with minimum training losses of 0.2485 and 0.2441 and validation losses of 0.2339 and 0.2821, respectively, highlighting their capacity to capture relevant relationships in the data while maintaining reliable performance.

Graph-based models GraphGCN and GraphSAGE also exhibited stable convergence, albeit with slightly higher final losses compared to the hypergraph-based models. Nevertheless, they maintained consistent trends with no abrupt oscillations or divergence. These results suggest that while all models effectively learned from the data, the hypergraph-based architectures might be better suited to capturing higher-order interactions. Overall, the smooth training and validation loss curves across all models indicate that the training setups were appropriately configured, resulting in robust learning with minimal risk of overfitting.

4.3.3 Running Time Analysis: The training running times of the models, available in Figure 4, reflect a combination of computational complexity and

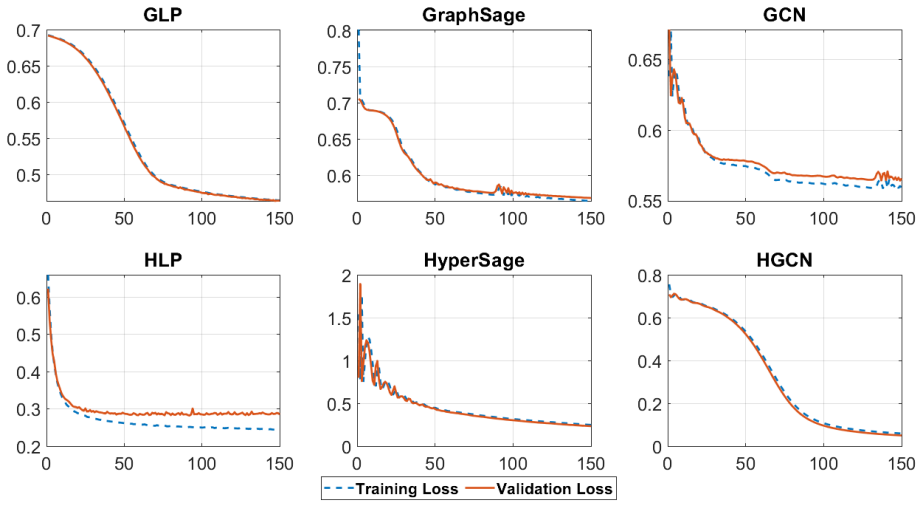

Fig. 3: Training loss and validation loss (y-axis) and the number of epochs (x-axis) for both graph and hypergraph models.

data representation. Overall, hypergraph-based models are generally faster than graph-based ones despite their ability to capture more complex relationships. This can be explained by the significant reduction in the number of edges in the hypergraph representation: while the graph contains over 780,000 edges, the hypergraph includes only 3,849 hyperedges, each representing a many-to-many relationship among concepts. This compression of structure significantly reduces the number of computations required during message passing and aggregation.

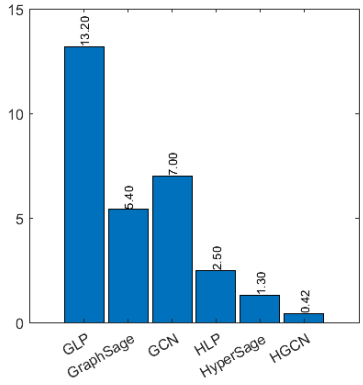

Fig. 4: The training running times (minutes) of the link prediction models.

Specifically, HGCN, the fastest model (0.42 minutes), benefits from both its spectrally normalized Laplacian formulation and the sparse hyperedge incidence matrix, enabling efficient batch processing over fewer structures. HyperSAGE (1.3 minutes) also takes advantage of the compact hypergraph structure but involves more steps per layer due to alternating node-hyperedge aggregations, resulting in slightly longer runtimes. HLP (2.5 minutes) is slower because it precomputes embeddings and performs dense feedforward operations on pooled vectors for each candidate group, a process that scales with the number of training examples rather than the hypergraph sparsity.

In contrast, GLP, GraphSAGE, and GCN operate on the full graph with over three-quarters of a million edges. These models must repeatedly perform aggregation over very high-degree nodes (average degree $\sim$ 291, max degree = 3,248), leading to much higher computational overhead, especially for deep architectures or models with dynamic neighbor sampling. This explains their slower runtimes, with GLP taking the longest at 13.2 minutes.

5 Conclusion

In this paper, we investigate the potential of hypergraph-based neural architectures for predicting emergent trends within scientific literature. By modeling concept co-occurrences as hyperedges, we first represent higher-order relations between research articles as a hypergraph. We later build three hypergraph deep learning models, HLP, HyperSage, and HyperGCN, to capture both local and global relationships inherent in academic publications and use these models to predict keywords of future research articles. We further compare our hypergraph models with three baseline graph models, GLP, GraphSAGE, and GCN.

Our experimental results demonstrate that hypergraph-based models consistently outperform their graph-based counterparts across all key metrics, showcasing its ability to model higher-order interactions effectively. These findings underscore the importance of representing multi-concept relationships in scholarly networks to enhance predictive performance and trend forecasting. Beyond model performance, the study also highlights the computational advantages of hypergraph representations, which reduce structural redundancy and enable efficient message passing compared to densely connected graphs. This makes hypergraph-based approaches not only more accurate but also more scalable for large-scale bibliometric analyses.

For future work, integrating temporal dynamics into the hypergraph architecture remains a promising avenue. In particular, coupling hypergraph embeddings with temporal models such as HNN-LSTM could capture time-evolving patterns of research trends and further enhance forecasting capabilities. Additionally, expanding the study to include interdisciplinary datasets could validate the generalizability of these models across diverse scientific domains.

Bibliography

[1] Aggarwal, C.C., Wang, H.: Managing and mining graph data, vol. 40. Springer (2010)

[2] Arya, B., Yadav, P., Talukdar, P.: Hypersage: Generalizing graphsage to hypergraphs. arXiv preprint arXiv:2302.11648 (2023)

[3] Bornmann, L., Mutz, R.: Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. Journal of the association for information science and technology **66**(11), 2215–2222 (2015)

[4] Boyack, K.W., Klavans, R., Börner, K.: Mapping the structure of science. Nature **430**(6998), 631–634 (2005)
[5] Bretto, A.: Hypergraph theory: An introduction. Springer (2013)
[6] Bu, J., Tan, S., Chen, C., Wang, C., Wu, H., Zhang, L., He, X.: Music recommendation by unified hypergraph: combining social media information and music content. In: Proceedings of the 18th ACM international conference on Multimedia. pp. 391–400 (2010)
[7] Chien, E., Zhao, L., Wang, Y., Karypis, G., Akoglu, L.: You should evaluate your graph neural networks on hypergraphs. In: International Conference on Learning Representations (ICLR) (2022)
[8] Cook, D.J., Holder, L.B.: Mining Graph Data. John Wiley & Sons (2006)
[9] Feng, Y., You, H., Zhang, Z., Ji, R., Gao, Y.: Hypergraph neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 3558–3565 (2019)
[10] Fortunato, S., Bergstrom, C.T., Börner, K., Evans, J.A., Helbing, D., Milojević, S., Petersen, A.M., Radicchi, F., Sinatra, R., Uzzi, B., Vespignani, A., Waltman, L., Wang, D., Barabási, A.L.: Science of science. Science **359**(6379), eaao0185 (2018). https://doi.org/10.1126/science.aao0185, `https://www.science.org/doi/abs/10.1126/science.aao0185`
[11] Gipp, B., Beel, J., Hentschel, C.: Citation-based document retrieval for scientific publications using co-citation proximity analysis. Journal of Informetrics **14**(1), 101003 (2020)
[12] Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 855–864. ACM (2016)
[13] Hamilton, W.L., Ying, R., Leskovec, J.: Inductive representation learning on large graphs. In: Advances in neural information processing systems. vol. 30 (2017)
[14] Kedrick, K., Levitskaya, E., Funk, R.J.: Conceptual structure and the growth of scientific knowledge. Nature Human Behaviour **8**(10), 1915–1923 (2024)
[15] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
[16] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
[17] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. International Conference on Learning Representations (ICLR) (2017), `https://arxiv.org/abs/1609.02907`
[18] Krenn, M., Zeilinger, A.: Predicting research trends with semantic and neural networks. Nature **588**(7836), 201–204 (2020)
[19] Merris, R.: Laplacian matrices of graphs: a survey. Linear algebra and its applications **197**, 143–176 (1994)
[20] Mohar, B.: Some applications of laplace eigenvalues of graphs. In: Graph symmetry, pp. 225–275. Springer (1997)
[21] Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. Proceedings of the 27th international conference on machine learning (ICML-10) pp. 807–814 (2010)

[22] Ofer, D., Linial, M.: Emerging scientific topics predicted by combining text semantics and citation indicators. Patterns **4**(3), 100644 (2023)
[23] Praggastis, B., Aksoy, S., Arendt, D., Bonicillo, M., Joslyn, C., Purvine, E., Shapiro, M., Yun, J.Y.: Hypernetx: A python package for modeling complex network data as hypergraphs. Journal of Open Source Software **9**(95), 6016 (2024). https://doi.org/10.21105/joss.06016, `https://doi.org/10.21105/joss.06016`
[24] Saifuddin, K.M., Bumgardner, B., Tanvir, F., Akbas, E.: Hygnn: Drug-drug interaction prediction via hypergraph neural network. In: 2023 IEEE 39th International Conference on Data Engineering (ICDE). pp. 1503–1516. IEEE (2023)
[25] Sawhney, R., Agarwal, S., Wadhwa, A., Shah, R.R.: Spatiotemporal hypergraph convolution network for stock movement forecasting. In: 2020 IEEE International Conference on Data Mining (ICDM). pp. 482–491. IEEE (2020)
[26] Small, H.: Co-citation in the scientific literature: A new measure of the relationship between two documents. Journal of the American Society for Information Science **24**(4), 265–269 (1973)
[27] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. Journal of machine learning research **15**(1), 1929–1958 (2014)
[28] Wu, L., Wang, D., Evans, J.A.: Large teams develop and small teams disrupt science and technology. Nature **566**(7744), 378–382 (2019)
[29] Yadati, N., Nimishakavi, M., Yadav, P., Ajit, P., Chandra, S., Talukdar, P.: Hypergcn: A new method of training graph convolutional networks on hypergraphs. In: Advances in Neural Information Processing Systems. vol. 32 (2019)
[30] Yang, D., Qu, B., Yang, J., Cudre-Mauroux, P.: Revisiting user mobility and social relationships in lbsns: A hypergraph embedding approach. In: The world wide web conference. pp. 2147–2157 (2019)
[31] Zhang, M., Chen, Y.: Link prediction based on graph neural networks. In: Advances in Neural Information Processing Systems. vol. 31 (2018)
[32] Zheng, Y., Zhang, J., Wang, L., Li, X., Wu, J.: Forecasting scientific topic trends with spatio-temporal graph neural networks. Scientometrics **127**, 7415–7435 (2022)
[33] Zhou, D., Huang, J., Schölkopf, B.: Learning with hypergraphs: Clustering, classification, and embedding. In: Advances in Neural Information Processing Systems (NeurIPS) (2007)
[34] Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., Sun, M.: Graph neural networks: A review of methods and applications. AI Open **1**, 57–81 (2020)