EAI Endorsed Transactions
on Scalable Information Systems
Research Article EAI.EU

Team structure optimization and talent flow prediction based on graph neural network

Shuangshuang Chen[1] , Huaqiang Lai[1,*]

[1]Business School, Jiangsu Open University,No. 399 Jiangdong North Road, Gulou District, Nanjing, China; Jiangsu Innovation Ecology Research Institute, No. 399 Jiangdong North Road, Gulou District, Nanjing, China

Abstract

This paper proposes a graph neural network-based framework for team structure optimization and talent flow prediction to address the limitations of traditional methods in modeling complex organizational dynamics. By constructing an organizational graph structure containing 12,547 employees and 56,892 flow records, we develop a hybrid model combining graph attention networks and multi-layer perceptrons to accurately capture team collaboration relationships and talent mobility patterns. The proposed method achieves an F1 score of 78.6% in teamwork prediction, and 81.2% accuracy with 0.864 AUC in talent flow prediction, outperforming traditional methods by over 10%. Experimental results demonstrate that graph neural networks can effectively model complex dependencies in organizational structures, providing data-driven decision support for team optimization and talent management.

Keywords: graph neural network, co-simulation, optimization of team structure, talent flow forecasting, organizational management, Attention mechanism.

Received on 28 November 2025, accepted on 23 April 2026, published on 04 May 2026

doi: 10.4108/eetsis.11146

*Corresponding author. Email: laihq@jsou.edu.cn

1. Introduction

With the rapid development of artificial intelligence technologies, graph neural networks (GNNs) have emerged as powerful tools for modeling complex relational data in various domains [1-3]. In organizational management, the ability to accurately model team structures and predict talent mobility patterns has become increasingly important for maintaining competitive advantage in dynamic business environments [4-5]. Traditional approaches to organizational analysis often rely on static statistical methods or simplified network models, which fail to capture the intricate dependencies and temporal dynamics inherent in real-world organizational structures [6-7].

Current research on organizational analytics primarily focuses on either structural optimization or mobility

prediction in isolation, with limited work addressing their synergistic relationship [8-9]. While graph-based methods have shown promise in social network analysis, their application to organizational contexts remains underexplored, particularly in capturing the complex interplay between team composition and individual career trajectories [10]. Furthermore, existing methods often overlook the temporal evolution of organizational networks and the multi-scale nature of team-talent interactions.

The research gap identified in this study lies in the lack of integrated frameworks that simultaneously address team structure optimization and talent flow prediction while accounting for their mutual influence. Most existing approaches treat these as separate problems, failing to leverage the rich information contained in their coupling. Additionally, there is a need for methods that can handle the scale and complexity of modern organizational data while providing interpretable insights for management decisions. Beyond its applications in large enterprises, Graph Neural Networks (GNNs) have recently been utilized to address specific challenges in small and medium-sized enterprises (SMEs). For instance, researchers have explored using GNNs to analyze risk propagation in SME supply chain networks [11], or to predict corporate credit risks by modeling lending relationships and guarantee networks among SMEs [12]. These studies demonstrate that GNNs can effectively handle the challenges of sparse data and complex relationships in SMEs, providing precise management insights for resource-constrained businesses. However, existing research primarily focuses on single business scenarios, lacking a comprehensive framework like the one proposed in this study that simultaneously optimizes internal team structures and predicts core talent attrition. This further highlights the theoretical and practical innovation value of our research.

Focusing on the core issues of team structure optimization and talent flow prediction, this paper systematically constructs a theoretical framework and method system based on graph neural networks. The research content mainly includes three dimensions: first, establish a graph representation model of team structure, abstract team members as nodes, abstract collaborative relationships as edges, and capture complex dependencies within the team through graph convolutional networks and attention mechanisms; Secondly, the prediction model of talent flow is designed, and the dynamic graph neural network is used to analyze the talent flow pattern in the organizational network to achieve accurate prediction of future flow trends. Finally, a collaborative simulation platform for team structure and talent flow is developed, and the multi-agent system and discrete event simulation technology are integrated to verify the implementation effect of different management strategies. Through these three levels of in-depth research, this paper aims to provide a data-driven intelligent decision support system for organizational management and promote the development of organizational management in the direction of refinement and intelligence.

2. Team structure optimization and talent flow prediction algorithm

2.1. Graph neural network model

A graph is a data structure that models a set of objects and their relationships between them. With its powerful ability to express various systems in reality, the research direction of using machine learning and deep learning methods to analyze graph structure data has attracted more and more attention, and related research spans many fields such as social sciences, natural sciences, and knowledge graphs [11]. Among them, node classification, link prediction and clustering are all mainstream directions of graph analysis research. Because graph data presents non-Euclidean characteristics, it is completely different from the Euclidean data structure of traditional text and images, and the deep learning methods that have achieved excellent results in many fields in the past decade cannot be directly applied to graph structure data[12]. Therefore, how to adapt deep learning technology to the data and tasks of graph structure has attracted widespread attention and has become one of the hot research topics in recent years [13].

For graphs with variable topology, a general approach to processing structural information is required. In order to ensure universal applicability, graph structure information processing methods must be designed without a known fixed causal hypothesis. The treatment of local representation

solves this problem by building models at the node level rather than the full graph, which only cares about the relationships between nodes and their vicinity [14]. This processing can greatly reduce the number of parameters required for the model, similar to how CNN convolutional kernels are reused on pixels, and the processing of local representation can also be reused on all nodes, allowing for efficient combination of the "experience" of all nodes and graphs in the dataset to learn individual functions. Despite these advantages, local representation does not by itself solve the structural representation problem of variable neighborhood diagrams, as there is no consistent way to order nodes in a neighborhood. A common solution to this situation is to use a displacement invariant function (Equation 1 [15]) as the neighborhood information traverses for each node, and the output of the displacement invariant function does not change with changes in the order of the input elements. Because of this feature, it is ideal for handling any number of input elements, which will come in handy when dealing with topological variable disordered and non-positional graphs.

$$\Psi(Z) = \phi\left(\sum_{z \in Z} \psi(z)\right) \tag{1}$$

For graphs with cyclic relationships, it is necessary to deal with the mutual causal relationship between node states. Under the local processing assumption, the intermediate state of any node in the graph is a function of its neighbor state. When node state is computed in parallel, cyclic structural dependencies are transformed into intercausal dependencies, triggering potentially infinite loops. This problem can be solved by setting the iteration scheme to use nodes and ℓ calculate the state vl+1 at +1 iteration to calculate the state vl+1 from the lth iteration.

Context diffusion is arguably the most important concept in local graph learning methods. As the name suggests, the goal of context diffusion is to disseminate information across the graph, providing nodes with knowledge about them in a wider context rather than being limited to their immediate neighborhood, so that node feature representations can be better generated[16]. Practice shows that $\ell = 3$ iterations is sufficient to increase the contextual information including all nodes in the graph.

According to different context diffusion mechanisms, most graph learning models can be divided into three types of architecture: loop, feedforward and constructive. The Recurrent Architectures model regards the iterative processing of node information as a dynamic process, and the representative models of this type of model are graph neural networks [17] and graph echo state networks [18], which use single-layer loop units to model the interdependencies between node states to process graph loops by constraining the dynamic convergence process of the model. Among them, graph neural network is the earliest semi-supervised deep learning method for graph data.

Instead of an iterative diffusion mechanism on the same layer of a recursive unit, the feedforward architecture model stacks multiple layers to form a local context that learns in each iteration, and the interdependencies caused by loops are managed through layers with different parameters without constraints on the coding process to guarantee convergence. The feedforward architecture model is popular for its simplicity and efficiency on many tasks. However, deep graph neural networks face gradient-related problems such as gradient disappearance and gradient explosion as other deep neural networks, especially when learning is done "end-to-end" throughout the architecture.

Constructive architecture models can be regarded as special cases of feedforward models, with representative models such as NN4G[19] and CGMM[20]. The constructive architecture model constructs a deep learning model framework that can process graph structure data by recursively and repeatedly combining some basic computing units such as convolution and attention. The main benefit of this architecture is that the deep network does not cause vanishing/exploding gradient issues due to design, so context can be better propagated between nodes. In addition, another important feature of this architecture is the "divide and conquer" approach to solving problems, gradually splitting tasks into simpler subtasks and thus relaxing, solving a sub-problem at one level, and gradually solving the global task using the results of the previous layer.

The core of local graph processing is to aggregate the neighborhood of the target node to compute the node representation. Following the general assumption that the nodes in the graph are out of order, a displacement invariant function needs to be used to implement the aggregation

process. The neighborhood aggregation function of node v at the L+1 layer can be expressed as:

$$\mathbf{h}_v^{\ell+1} = \phi^{\ell+1}\left(\mathbf{h}_v^{\ell}, \Psi\left(\left\{\psi^{\ell+1}\left(\mathbf{h}_u^{\ell}\right) | \ u \in \mathsf{N}_v\right\}\right)\right) \tag{2}$$

Field N It can be open or closed, Ψ represents the substitution invariant function, $\ell = 0$ Corresponding to node feature x, some kind of nonlinear transformation that does not depend on structural information. For example, the neighborhood aggregation function in a graph convolutional network can be expressed as:

$$\mathbf{h}_v^{\ell+1} = \sigma\left(\mathbf{W}^{\ell+1} \sum_{u \in \mathsf{N}(v)} \mathbf{L}_{uv} \mathbf{h}_{uu}^{\ell}\right) \tag{3}$$

where L is the normalized Laplacian operator, W is the weight matrix, and σ is the nonlinear activation function.

The general neighborhood aggregation scheme mentioned above requires edges to be non-attributed or belong to the same property, which is usually not true, because the edges of the graph most often contain information, which can be discrete or continuous. Therefore, we need to use the properties of edges to enrich the mechanism of node representation. We can reconstruct the neighborhood aggregation function according to the label ck of the edge, and the set A of the neighborhood edge corresponds to Equation 2.4 for the finite and discrete case, and the corresponding formula for the continuous case is Equation 2.5, as follows:

$$\mathbf{h}_v^{\ell+1} = \phi^{\ell+1}\left(\mathbf{h}_v^{\ell}, \sum_{c_k \in \mathsf{A}} \left(\Psi\left(\left\{\psi^{\ell+1}\left(\mathbf{h}_u^{\ell}\right) | \ u \in \mathsf{N}_v^{c_k}\right\}\right) * w_{c_k}\right)\right) \tag{4}$$

$$\mathbf{h}_v^{\ell+1} = \phi^{\ell+1}\left(\mathbf{h}_v^{\ell}, \Psi\left(\left\{e^{\ell+1}\left(\mathbf{a}_{uv}\right)^T \psi^{\ell+1}\left(\mathbf{h}_u^{\ell}\right) | \ u \in \mathsf{N}_v\right\}\right)\right) \tag{5}$$

For cases where the edge labels are continuous, e in the formula can be any function.

The basic idea of graph neural network is to iteratively update the representation of nodes by aggregating neighbor nodes and their own representations.

$$a_v^k = \text{Aggregate}^{\,k}\left\{H_u^{k-1} : u \in N(v)\right\} \tag{6}$$

$$H_v^k = \text{Combine}^{\,k}\left\{H_v^{k-1}, a_v^k\right\} \tag{7}$$

Compared with traffic energy consumption data under normal conditions, charging demand data on icy and snowy roads exhibits stronger volatility and uncertainty, mainly reflected in the following aspects:

Under extreme weather conditions such as snowfall and icing, the travel behavior of electric vehicles is significantly affected. For example, during heavy snowfall, some vehicles are forced to suspend or delay travel, resulting in a noticeable drop in the demand curve; conversely, after road clearance, concentrated travel often leads to a sudden surge in demand. In icy and snowy conditions, vehicle energy consumption depends not only on travel distance but also on factors such as the road surface friction coefficient and battery performance degradation under low-temperature conditions. These factors cause significant short-term fluctuations in demand data, presenting non-stationary characteristics. In actual observed data, traffic sensors may be affected by snow cover or meteorological interference, leading to abnormal values; additionally, packet loss at some sampling points further increases the noise content in the data.

To address the above issues, this study introduces the Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) method to denoise the raw data. This method decomposes complex non-stationary signals into a set of intrinsic mode functions (IMFs) with physical meaning, and when combined with entropy analysis and wavelet thresholding effectively distinguishes trend signals from high-frequency noise, thereby achieving a smooth representation of charging demand data [21-22].

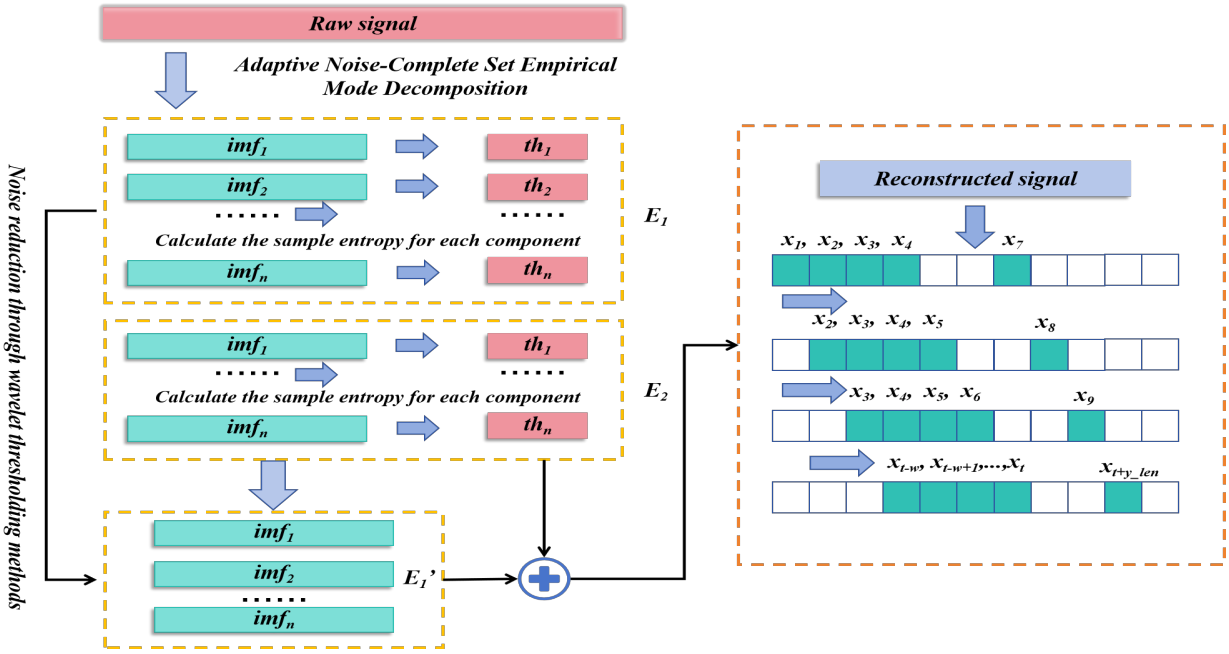


Figure 1. Data Denoising Module Flowchart

The overall workflow of the denoising module is illustrated in Figure 1. The proposed CEEMDAN-AGNN-GTCN hybrid model focuses its spatiotemporal modeling efforts primarily on the AGNN+GTCN module. In this framework, the AGNN module integrates a graph neural network with a multi-head attention mechanism.

In the Graphometry model, three types of encoding are employed: centrality encoding, spatial encoding, and edge encoding. These encodings fully capture the spatial structural features of graphs. Edge encoding and spatial encoding are embedded into the computations of the multi-head attention mechanism, while centrality encoding is incorporated into the input features of graph nodes. By embedding these encodings, the graph neural network enhanced with multi-head attention is able to effectively extract spatial features between electric vehicle (EV) charging stations, thereby optimizing the performance of the GNN in spatial information representation.

The temporal convolutional network module (GTCN) introduces a gated TCN structure, designed to extract temporal dependencies in EV charging demand. It consists of two TCNs, each connected to different activation functions, which enables effective control of the inflow of valid temporal features while discarding irrelevant ones. The TCN architecture primarily relies on dilated convolutions to process time-series data. Since causal convolutions are used, the model avoids recursive connections typical of RNNs, thereby accelerating the training speed in temporal dependency extraction [23].

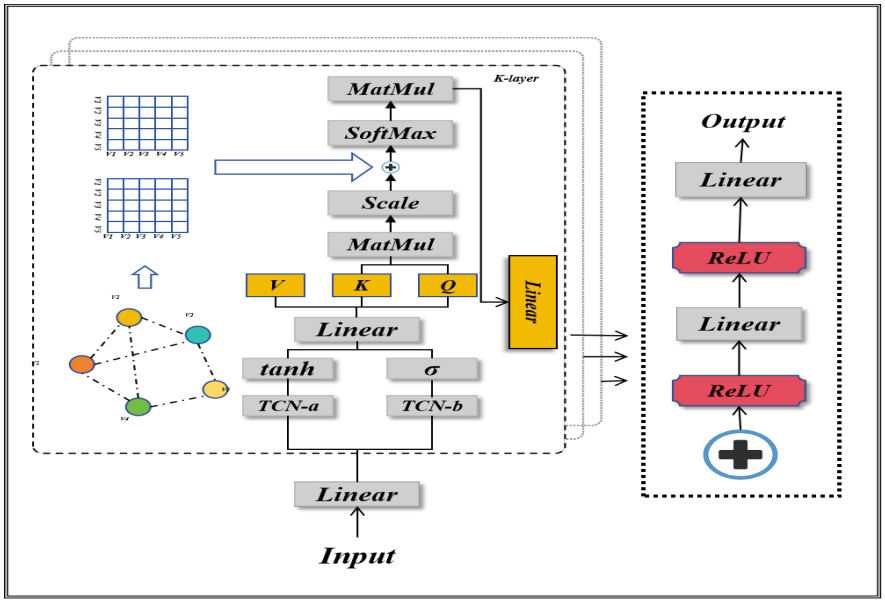


Figure 2. GNN+GTCN Module

The detailed architecture of the AGNN+GTCN module is illustrated in Figure 2. Temporal sequence data of the graph neural network first pass through a linear layer into the gated TCN module for temporal feature extraction. Subsequently, spatial features are captured through the graph neural network module enhanced with multi-head attention. After processing through K layers of spatiotemporal network blocks with residual skip connections, the model outputs the final sequence through two linear layers alternated with two nonlinear activation functions (Rectified Linear Unit, REL).

2.2. Model training

This section primarily employs implicit associations within heterogeneous team structure networks to train the proposed graph neural network.

To assess the accuracy of the generated cooperative and competitive preference vectors, the team structure relationships are partitioned into 60% for training, 10% for validation, and 30% for testing according to reference[24]. Notably, to simulate real-world scenarios where associations between partner and competitor companies are unknown, the dataset processing removes all edge connections between the training and test sets, thereby preventing information leakage during evaluation.

The model's performance is evaluated using two types of metrics. The first category consists of accuracy measures, which assess the correctness of company classification. These indicators provide an intuitive evaluation of prediction performance for unknown relationships and include precision, recall, F1 score, and the area under the ROC curve. The second category comprises ranking metrics. For each team, the experiment evaluates ranking effectiveness by comparing candidate output scores against actual relationship labels. Such metrics are particularly suitable for assessing real-world cooperative competition strategy analysis, such as recommending potential partners or rival companies. These ranking indicators include normalized discounted cumulative gain and mean average precision, both widely adopted in recommendation tasks.

All models are implemented using the deep learning framework Porch and the graph learning framework DGL, with parameters initialized via the Xavier method cited in references[25] and[26]. Regarding hyperparameter configuration, the bidirectional gated recurrent unit employs a single GRU layer, the multi-graph convolutional network's RGCN uses two layers, and the embedding vector dimension

is set to 50. During training, the Adam optimizer referenced as is applied for parameter optimization, with a learning rate of 0.01 and a batch size of 32.

To validate the accuracy of the learned preference vectors, this study compares prediction performance against several state-of-the-art cooperation and competition relationship prediction algorithms. Baseline methods are divided into two groups: the first relies on recommendation system techniques to suggest potential partners or competitors for each company, while the second utilizes company embedding representations to predict possible inter-company relationships. Further details of these baseline approaches are provided in Figure 3.

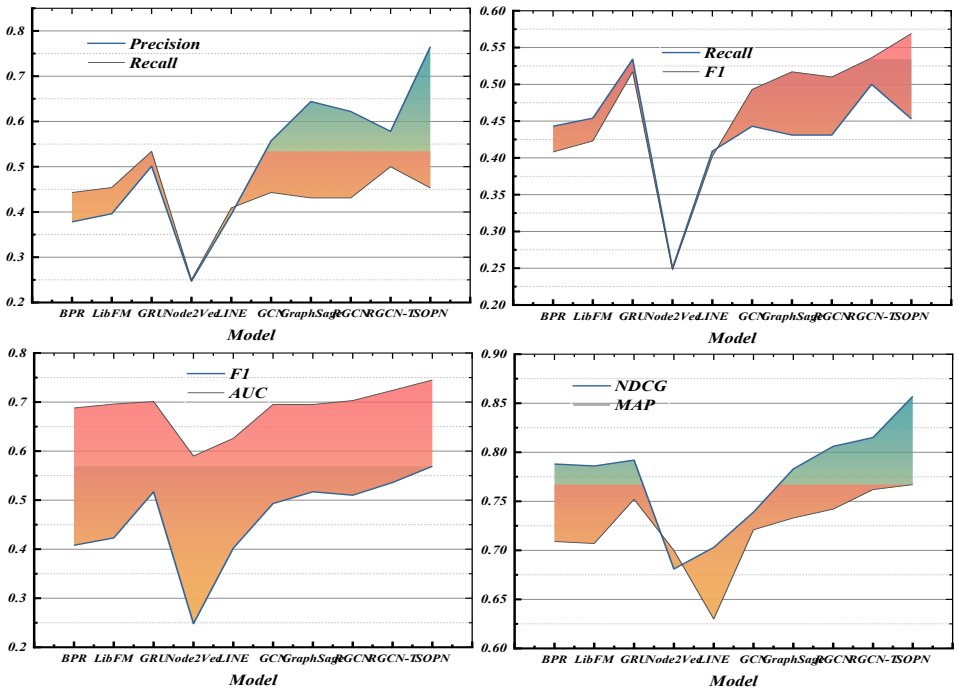


Figure 3. Experimental results on the partnership prediction task

Figure 4 shows the simulation results of the method based on the recommendation system. BPR is a recommendation method based on implicit feedback. Adapted to the experimental setup of this paper, the observed samples of cooperation and competition can be regarded as positive feedback, respectively.

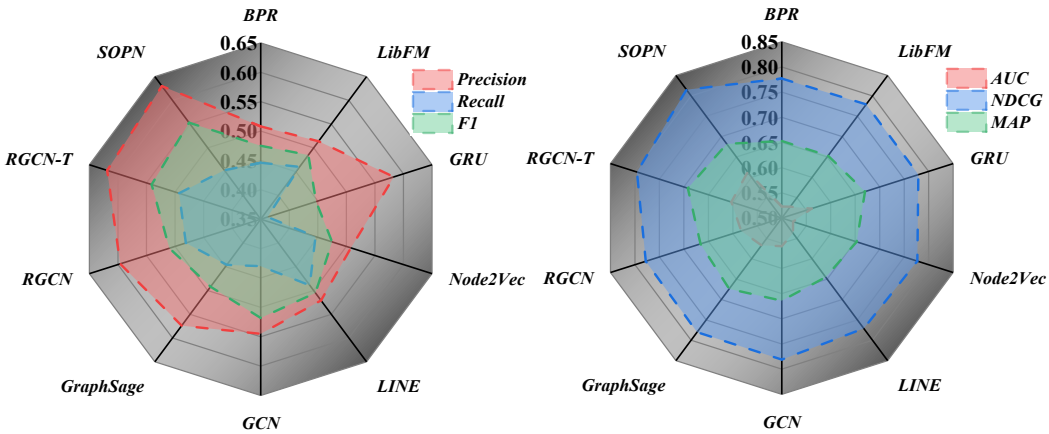


Figure 4. Experimental results on the competition prediction task

Figure 5 shows the prediction effect of the network model compared with the baseline method in the prediction of cooperation and competition. As mentioned above, the prediction effect is measured from two aspects: prediction accuracy indicators and ranking indicators. From the comparison of experimental results, the following conclusions can be drawn: First, the performance of the symbolic preference network in the two tasks of cooperation relationship prediction and competitive relationship prediction exceeds the most advanced baseline comparison method in most evaluation indicators. This verifies that the preference vector generated by the symbolic preference network model proposed in this paper can accurately summarize the company's preference for cooperative companies and competing companies and then predict the cooperation and competition relationship more accurately. Second, it is not difficult to see from the comparison between the three methods of GRU, GCN, and RGCN that different forms of input have different effects on the final prediction performance, and combining more inputs can achieve better accuracy. This validates the necessity of handling heterogeneous inputs. Thirdly, the symbolic preference network proposed in this paper has more advantages in predicting cooperative companies, which indicates that the method is more helpful in mining potential cooperative partners for companies, which have certain guiding value for the application in real scenarios. This suggests that partnership predictions rely more on text input, while competitive relationship predictions rely more on relationship information between companies. For the proposed method, both text and relational inputs are effectively modeled, so they can perform better in both cooperation and competition prediction tasks.

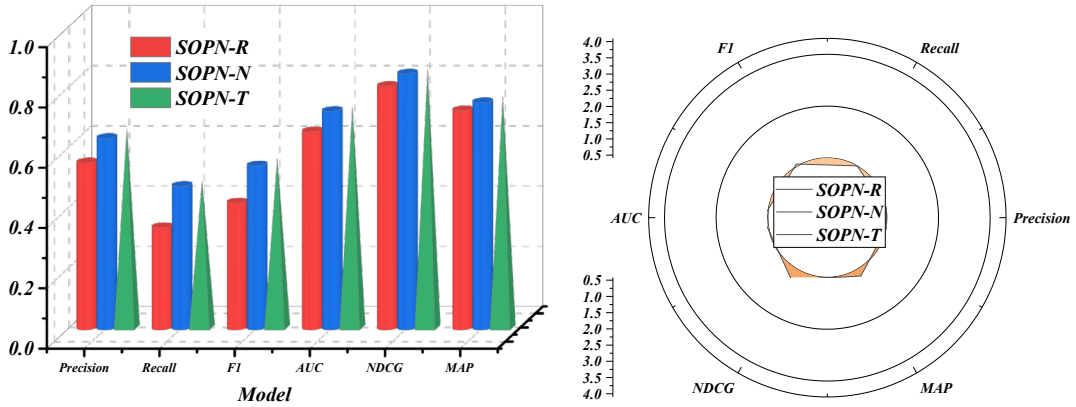


Figure 5. Model ablation experiment on the prediction task of cooperation and competition

3. Team structure optimization simulation

3.1. Team structure optimization simulation method

Team structure optimization is one of the core issues in organizational management, and its goal is to adjust the collaborative relationship and task allocation mechanism between team members through scientific methods, so as to improve the overall performance of the team[27]. Team structure optimization aims to adjust collaborative relationships and task allocation through computational methods. By abstracting teams as graph structures where nodes represent members and edges represent collaborations, we establish a performance evaluation model based on graph neural networks. With the development of computational organization theory and artificial intelligence technology, the team structure optimization method based on graph neural network has gradually shown its unique advantages [28-29]. This method abstracts the team into a graph structure, where

nodes represent team members and edges represent collaborative relationships or task dependencies among members [30].

The core of team structure optimization lies in establishing a reasonable performance evaluation model and optimizing the objective function. Suppose a team consists of n members whose collaborative relationships can use an adjacency matrix $A \in \square^{n \times n}$ Show, in its $A_{ij} = 1$ Show, in it $x_i \in \square^{d}$, characterizing their skill level, experience background and behavioral preferences. Team performance P can be defined as a function of member characteristics and collaboration structures:

$$P = f(X, A) = \sigma(W \cdot \text{Aggregate}(X, A) + b) \quad (8)$$

thereinto $X = [x_1, x_2, \ldots, x_n]^T$ For the member feature matrix, Aggregate is the aggregation function in the graph neural network to integrate the information of neighbor nodes, W and b are the model parameters, and σ is the activation function. The model captures local and global dependencies among team members, allowing for a more accurate assessment of team performance in different task environments.

To optimize your team structure, you need to set clear optimization goals [31]. Common optimization goals include maximizing team performance, minimizing communication costs, and improving task completion efficiency. Taking maximizing team performance as an example, the following optimization problem can be constructed:

$$\max_A P = f(X, A) \quad \text{s.t.} \quad \sum_{i,j} A_{ij} \le B \quad (9)$$

where B is the maximum number of team collaboration relationships, reflecting the constraints of organizational resources. This optimization problem is a combinatorial optimization problem, which is difficult to solve efficiently by traditional methods. Through end-to-end training, graph neural networks can learn the optimal cooperation structure A^*. This enables dynamic optimization of team structure.

In addition, team structure optimization also needs to consider the degree of matching task characteristics. Suppose task T consists of a set of skill demand vectors $t \in \square^{d}$ Description, the team's fitness to the task can be defined as:

$$S(T, X, A) = \text{Sim}(t, \text{Pooling}(X, A)) \quad (10)$$

Pooling is a graph pooling operation that generates the feature representation of the team as a whole, and Sim is a similarity calculation function. By maximizing $S(T, X, A)$ to ensure that the team structure is highly matched with the tasks it undertakes, thereby improving the quality and efficiency of task completion.

3.2. Team collaboration optimization based on graph neural network

A measure of skill complementarity between member I and member j. This formula shows that the efficiency of collaboration not only depends on the intensity of collaboration but also is closely related to the skill match between members [32-33].

In order to optimize the efficiency of team collaboration, a graph neural network model needs to be built to learn the optimal edge power matrix W^*. As a typical representative of graph neural network, graph convolutional network can aggregate neighbor node information through multi-layer convolutional operations to generate embedded representations of nodes. The l-layer graph convolution operation can be expressed as:

$$E_{\text{team}} = \sum_{i=1}^{n} \sum_{j=1}^{n} W_{ij} \cdot \text{Sim}\left(x_i, x_j\right)$$

thereinto $\text{Sim}\left(x_i, x_j\right)$ A measure of skill complementarity between member I and member j. This formula shows that the efficiency of collaboration not only depends on the intensity of collaboration but also is closely related to the skill match between members.

In order to optimize the efficiency of team collaboration, a graph neural network model needs to be built to learn the optimal edge power matrix W^*. As a typical representative of graph neural network, graph convolutional network can aggregate neighbor node information through multi-layer

convolutional operations to generate embedded representations of nodes [34-35]. The l-layer graph convolution operation can be expressed as:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)} \right) \tag{11}$$

After obtaining the node embedding representation, the edge weight matrix can be predicted by the inner product operation:

$$\hat{W}_{ij} = \sigma\left(h_i^T \cdot h_j \right) \tag{12}$$

where h_i and h_j are the final embedding representations of member I and member j, respectively, and σ is the Sigmoid function for mapping the output to the [0,1] interval. By minimizing the mean square error between the predicted edge power and the real edge weight, the graph neural network model can be trained:

$$L = \frac{1}{|E|} \sum_{(i,j)\in E} \left(\hat{W}_{ij} - W_{ij} \right)^2 \tag{13}$$

After the training is completed, the model can be used to predict the strength of unknown collaborative relationships and provide a basis for team structure optimization.

To verify the effectiveness of the proposed method, we conducted experiments on the simulation team dataset. The dataset consists of 50 teams, each consisting of 5 to 10 members, whose characteristics include skill level, frequency of communication, and task completion history. We compare the proposed graph neural network approach with the baseline method, including the recommended method based on collaborative filtering versus the method based on traditional graph embedding. The experimental results show that the proposed method is better than the baseline method in the cooperative relationship prediction task, as shown in Figure 6.

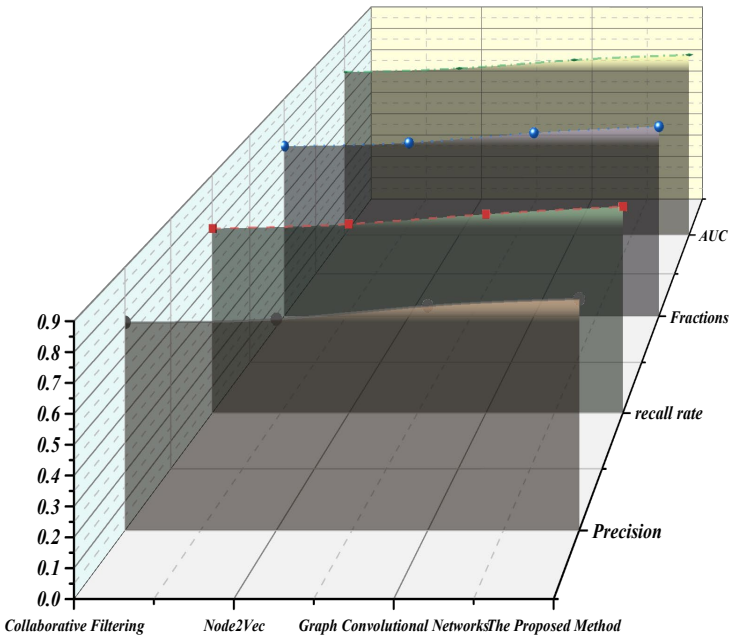


Figure 6. Comparison of predictive performance of team collaboration relationships

4. Construction of talent flow prediction model

4.1. Graph representation of talent flow characteristics

In organizational management research, talent flow is a

complex dynamic process involving the interaction of individual characteristics, organizational environment, social networks and other factors. In order to systematically characterize the internal law of talent flow, this paper uses a graph structure to model the characteristics of talent flow. Specifically, consider individuals in the organization as nodes in the graph and the flow relationships between individuals as edges, constructing a directed weighted graph G=(V,E,W), where V represents the set of all employees, E represents the flow path between employees, and W represents the weight of flow, usually determined by factors such as flow frequency, flow direction, or flow cost.

In order to capture the temporal characteristics of talent flow, this paper further introduces a dynamic graph model. Let G (t) represent the organizational state chart at time step t, and its nodes and edges may change over time. The construction of dynamic graphs allows us to analyze the evolutionary trends of talent mobility and provide a database for predicting future mobility. Specifically, we define a time window τ, dividing historical data into multiple consecutive time segments, each corresponding to a graph snapshot.

In terms of feature representation, in addition to basic node properties, structural features and contextual features are also introduced. Structural characteristics include graph indicators such as degree centrality, intermediate centrality, and clustering coefficient of nodes, which are used to measure the position and influence of individuals in the organizational network. Contextual features are described by the local neighborhood information of nodes, such as the attribute distribution of neighbor nodes, the weight distribution of edges, etc. In order to fuse these multi-source features, we design a composite feature vector with the following formula:

$$\mathbf{h}_i = \sigma\left(\mathbf{W}_s \cdot \mathbf{f}_s(i) + \mathbf{W}_c \cdot \mathbf{f}_c(i) + \mathbf{b}\right) \tag{14}$$

$\mathrm{f}_s(i)$ and $\mathrm{f}_c(i)$ Represents the structural features and contextual features of node I, respectively. In this way, we can organically combine the static attributes of talents with dynamic network relationships to provide rich feature inputs for subsequent flow predictions.

4.2. Talent flow prediction based on graph neural network

Based on the graph representation method mentioned above, this paper uses graph neural network to predict talent flow. GNNs can aggregate information about neighboring nodes through messaging mechanisms, thereby capturing complex dependencies between nodes. Specifically, we use graph convolutional networks as the base model, and the interlayer propagation rules are as follows:

$$\mathbf{H}^{(l+1)} = \sigma\left(\hat{\mathbf{D}}^{-\frac{1}{2}}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\right) \tag{15}$$

thereinto $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is an adjacent matrix that joins the self-ring. Through multi-layer graph convolution operations, each node's representation is able to blend information from its multi-hop neighbors, providing a more comprehensive representation of its state in the organization's network.

In order to further improve the prediction performance, this paper introduces an attention mechanism to construct a graph attention network. GAT allows nodes to assign different weights when aggregating neighbor information, providing more flexibility in capturing key influence factors.

In talent mobility prediction scenarios, the attention coefficient αij can be intuitively interpreted as the influence weight of employee j on employee i's departure decision. When two employees frequently collaborate within the organizational network (i.e., when an edge exists between them) and their characteristics are similar or complementary, the model learns a higher αij, indicating that their mobility decisions may influence each other.Its core calculation formula is:

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\sum_{j \in \mathrm{N}(i)} \alpha_{ij}\mathbf{W}^{(l)}\mathbf{h}_j^{(l)}\right) \tag{16}$$

α_{ij} Represents the attention coefficient of node i towards node j, calculated as follows:

$$\alpha_{ij} = \frac{\exp\left(\text{LeakyReLU}\left(\mathbf{a}^T\left[\mathbf{W}\mathbf{h}_i \| \mathbf{W}\mathbf{h}_j\right]\right)\right)}{\sum_{k\in N(i)} \exp\left(\text{LeakyReLU}\left(\mathbf{a}^T\left[\mathbf{W}\mathbf{h}_i \| \mathbf{W}\mathbf{h}_k\right]\right)\right)} \quad (17)$$

Here, a denotes the learnable parameter vector of the attention mechanism, and // represents the vector concatenation operation. By incorporating the attention mechanism, the model can adaptively focus on neighboring nodes that exert a greater influence on the current node, thereby enhancing prediction accuracy.

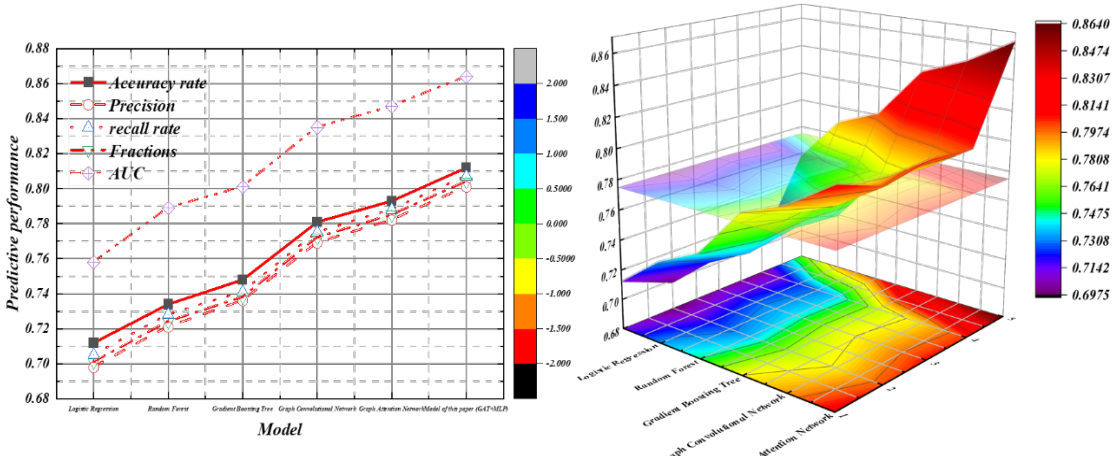
At the output layer, we employ multi-layer perception to map the final representations of nodes into flow probabilities. Specifically, for each node i, its outflow probability p_i^{out} inflow probability p_i^{in} The calculations are as follows:

$$p_i^{\text{out}} = \text{sigmoid}\left(\mathbf{W}_{\text{out}}\mathbf{h}_i^{(L)} + b_{\text{out}}\right), \quad p_i^{\text{in}} = \text{sigmoid}\left(\mathbf{W}_{\text{in}}\mathbf{h}_i^{(L)} + b_{\text{in}}\right) (18)$$

4.3.Predictive Model Performance Evaluation

To comprehensively evaluate the performance of the proposed talent mobility prediction model, this paper designed multiple experimental sets and analyzed them across three dimensions: accuracy, robustness, and interpretability. The experimental data were sourced from the human resources records of a major technology enterprise spanning 2018 to 2023, encompassing 12,547 employees and 56,892 mobility records. The data were partitioned into training, validation, and test sets at ratios of 60%, 20%, and 20% respectively. All experiments were replicated five times, with the average serving as the final result.To ensure data quality during dataset construction, we implemented the following screening criteria: (1) Employee tenure: Only retained employees with at least 6 months of active employment records within the study period (2018-2023) were included to exclude noise from temporary or short-term interns. (2) Collaboration validity: Team collaborations were defined as pairs of employees who completed over 5 joint tasks in project management systems or demonstrated monthly interaction frequency exceeding departmental averages in collaborative software, ensuring constructed graph edges represented meaningful and sustained working relationships. (3) Mobility record completeness: Each talent mobility record (including internal transfers and departures) required explicit effective dates and departmental/status flow directions, with incomplete records excluded. After these screenings, we obtained 12,547 valid employees and 56,892 valid mobility records from the raw data.

Figure 7 illustrates the predictive performance of different models on the test set. The results demonstrate that the proposed hybrid model, combining a graph attention network with a multi-layer perception, outperforms the comparison methods across all metrics. Regarding precision, recall, and F1 score, the proposed model achieved 80.1%, 80.7%, and 80.4% respectively, demonstrating robust overall performance. Furthermore, the model's AUC value of 0.864 indicates strong classification discrimination capability. These findings validate the advantages of graph neural networks in capturing complex talent mobility patterns, alongside the positive impact of incorporating attention mechanisms on enhancing predictive accuracy.



(1) Predictive performance comparison

(2) Corresponding three-dimensional diagram

Figure 7. Comparison of the predictive performance of the model on the test set

Figure 8 Effects of different feature combinations on model performance. When only the basic attributes of employees (such as age, position, performance, etc.) are used, the accuracy of the model is 73.5%, which is relatively poor. After adding structural features (such as degree centrality and intermediate centrality, etc.), the accuracy is increased to 78.4%, indicating that network structure information is of great value for talent flow prediction. After further adding contextual features (such as neighbor attribute distribution, edge weight, etc.), the accuracy is further improved to 79.6%. Ultimately, when using all features, the model achieves optimal performance with an accurate rate of 81.2%. These results show that talent flow is influenced by many factors, and more accurate predictions can only be achieved by comprehensively considering individual attributes, network structure, and local context.

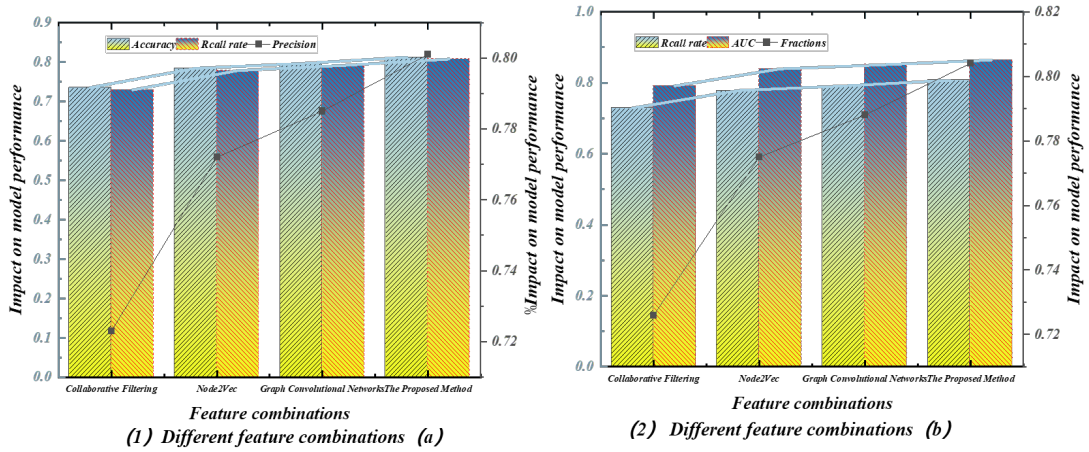


Figure 8. Effects of different feature combinations on model performance

Figure 9(1) Predictive performance in different departments. It can be seen from the results that the model performed best in the R&D department, with an accuracy rate of 82.5%, which may be due to the relatively stable personnel turnover pattern of the R&D department and the strong correlation between employee attributes and mobility behavior. The model maintains high prediction performance in all departments, indicating that it has good generalization ability. Figure 9(2) shows the impact of different time windows on prediction performance. When the time window is 3 months, the accuracy of the model is only 75.2%, indicating that short-term data is difficult to capture the long-term pattern of talent flow. The 12-month time window can better balance the sufficiency of the data and the sensitivity of the model, so this setting is used in subsequent experiments.

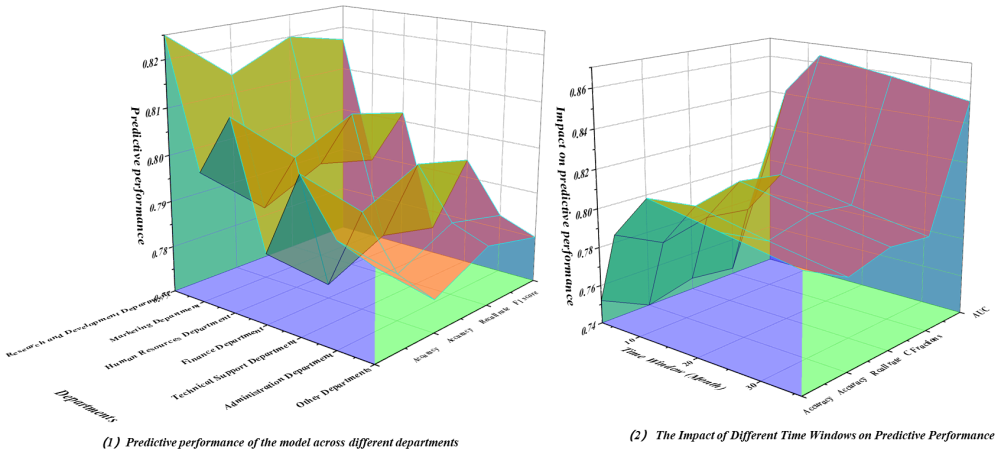


Figure 9. Predicted performance of the model in different departments

The proposed hybrid model combining graph attention network with multi-layer perceptron achieves superior performance across all evaluation metrics, with an accuracy of 81.2% and AUC of 0.864, representing a 10% improvement over conventional machine learning methods.

5. Collaborative simulation of team structure optimization and talent flow

5.1. The Mechanism of Which Team Structure Influences Talent Mobility

Team structure, as a crucial vehicle for relationships among members within an organization, exerts a profound influence on talent mobility behavior. Team structure, as a crucial representation of relationships among members within an organization, significantly influences talent mobility patterns through its impact on information flow and resource allocation efficiency.To delve into the mechanisms through which team structure influences talent mobility, we have constructed a dynamic analytical model based on graph neural networks. This model abstracts team structure as a graph, where nodes represent team members and edges denote collaborative relationships between them. By analyzing connection patterns, centrality metrics, and community structure characteristics among nodes, we uncover the intrinsic linkages between team structural features and talent mobility behavior.

In examining the influence of team structure on talent mobility, we introduce the concept of structural embedding vectors. Structural embedding vectors capture each member's positional characteristics within the team network, encompassing metrics such as degree centrality, closeness centrality, and betweenness centrality. These structural features are learned through graph neural networks to form structural representations for each member. Specifically, we employ graph convolutional networks for multi-layer feature extraction of team structures, where the feature extraction process at each layer can be represented as:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)} \right) \quad (19)$$

Beyond structural characteristics, we also examined the impact of functional complementarity among team members on talent mobility. Functional complementarity reflects the degree of variation in skills, expertise, and working styles among team members. We employed a diversity metric to quantify functional complementarity:

$$\text{Diversity}(v) = \frac{1}{|N(v)|} \sum_{u \in N(v)} \|x_v - x_u\|^2 \quad (20)$$

Among which $N(v)$ Indicates a node v the set of neighboring nodes, x_v x_u The eigenvectors of nodes v and u respectively. A higher value for this metric indicates greater skill disparity between the member and their peers, suggesting stronger functional complementarity. Our research reveals that moderate functional complementarity enhances team performance and reduces talent attrition, whereas excessive complementarity may increase communication costs, thereby exacerbating staff turnover.

The stability of team structure also constitutes a critical factor influencing talent mobility. We have defined a structural stability metric to measure the degree of change in team networks over time:

$$\text{Stability}\left(G_t, G_{t+1}\right) = 1 - \frac{\left|E_t \Delta E_{t+1}\right|}{\left|E_t \cup E_{t+1}\right|} \quad (21)$$

The closer this metric approaches 1, the more stable the team structure is. Through long-term tracking and observation, we have discovered a significant negative correlation between structural stability and talent attrition rates. That is to say, teams with more stable structures exhibit a lower likelihood of member attrition. This finding provides a theoretical basis for organizations to reduce talent loss by optimizing team structures.

5.2. Multi-Agent-Based Collaborative Simulation Platform

To investigate the dynamic interplay between team structure optimization and talent mobility, we designed and developed a multi-agent collaborative simulation platform. "The multi-

agent collaborative simulation platform integrates graph neural networks with discrete event simulation techniques to model talent mobility behaviors. The platform comprises four core modules: Environmental Management, Agent Behavior, Interaction Mechanism, and Data Analysis, which operate synergistically to simulate organizational dynamics."The platform architecture comprises four core modules: the Environmental Management Module, the Agent Behavior Module, the Interaction Mechanism Module, and the Data Analysis Module. These modules operate synergistically to form a comprehensive collaborative simulation ecosystem.

The Environmental Management Module is responsible for defining and maintaining macro-level organizational parameters, including organizational structure, resource allocation, market conditions, and industry trends. Employing a hierarchical design, this module features an organizational-level environment at the highest tier, a departmental-level environment at the intermediate tier, and a team-level environment at the lowest tier. Each environmental layer possesses specific parameter configurations and update rules, with bidirectional influence between layers achieved through information transmission mechanisms. Dynamic shifts in environmental parameters directly influence agent decision-making, thereby shaping talent mobility patterns.

The Agent Behavior Module constitutes the platform's core component, simulating diverse personnel behaviors within the organization. Each agent possesses autonomous decision-making capabilities, with behavioral choices grounded in a comprehensive assessment of internal state, environmental perception, and social influence. An agent's internal state encompasses psychological attributes such as competency level, career aspirations, job satisfaction, and organizational commitment. These attributes undergo dynamic shifts over time, shaping the agent's behavioral inclinations.

The Interaction Mechanism Module defines the rules governing interactions between agents and between agents and their environment. Employing an event-driven interaction model, relevant agents engage according to predefined rules when specific events occur. Interaction types encompass information exchange, resource allocation, task collaboration, and social interaction. To authentically simulate complex organizational relationships, we have designed a multi-layered interaction network incorporating both formal work relationship networks and informal social relationship networks.

5.3. Analysis of Collaborative Simulation Results

To validate the effectiveness of the collaborative simulation platform for optimizing team structures and talent mobility, we conducted a series of simulation experiments and collected extensive data for analysis. Figure 10 presents the correlation analysis results between different team structural characteristics and talent attrition rates. The table indicates a significant positive correlation between team size and talent attrition rate, suggesting that larger teams often face higher risks of talent loss. The clustering coefficient exhibits a negative correlation with talent attrition rate, suggesting that teams with stronger internal cohesion demonstrate greater member stability. The structural hole index shows a positive correlation with talent attrition rate, indicating that members occupying structural hole positions are more prone to departure. This may stem from their greater access to external opportunity information.

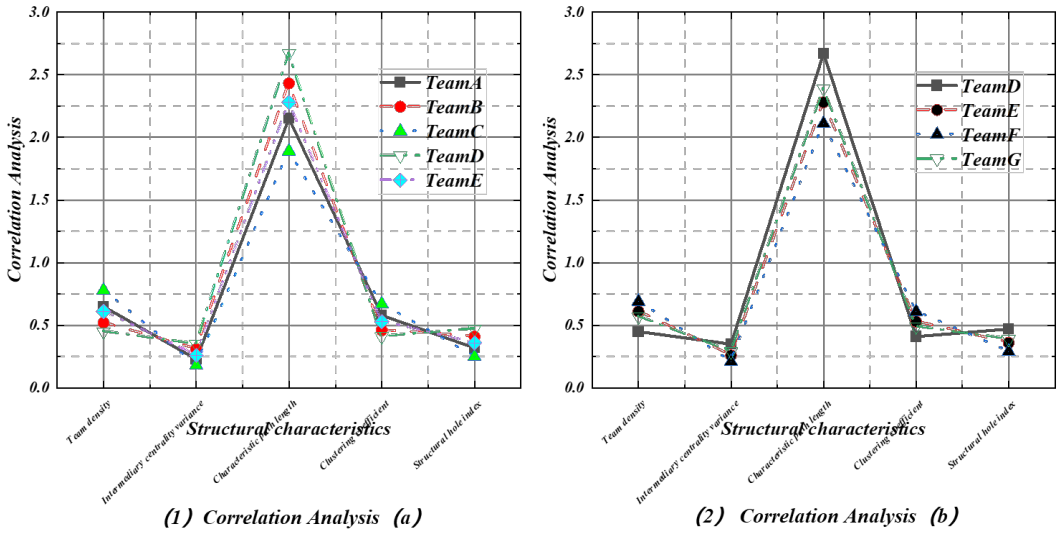


Figure 10. Correlation Analysis Between Team Structural Characteristics and Talent Turnover Rate

Figure 11(1) illustrates the impact of different intervention strategies on team structure, optimization and talent mobility. We compared four intervention approaches: random reorganization, competency-based optimisation, relationship-based optimization, and hybrid optimisation. As shown in the table, the hybrid optimisation strategy demonstrated the most favorable outcomes in reducing talent attrition and enhancing team performance, indicating that a team optimisation strategy integrating both competency matching and relationship quality proves most effective. Figure 11(2) illustrates the relationship between team structural characteristics and various dimensions of member satisfaction. Team density exhibits a positive correlation with all satisfaction dimensions, particularly showing the strongest correlation with interpersonal relationship satisfaction. This indicates that close-knit team connections enhance members' sense of belonging and overall satisfaction. Intermediary centrality variance exhibits negative correlations with all satisfaction dimensions, most strongly with growth opportunity satisfaction. This suggests that teams with excessive power concentration restrict members' development space, thereby diminishing satisfaction

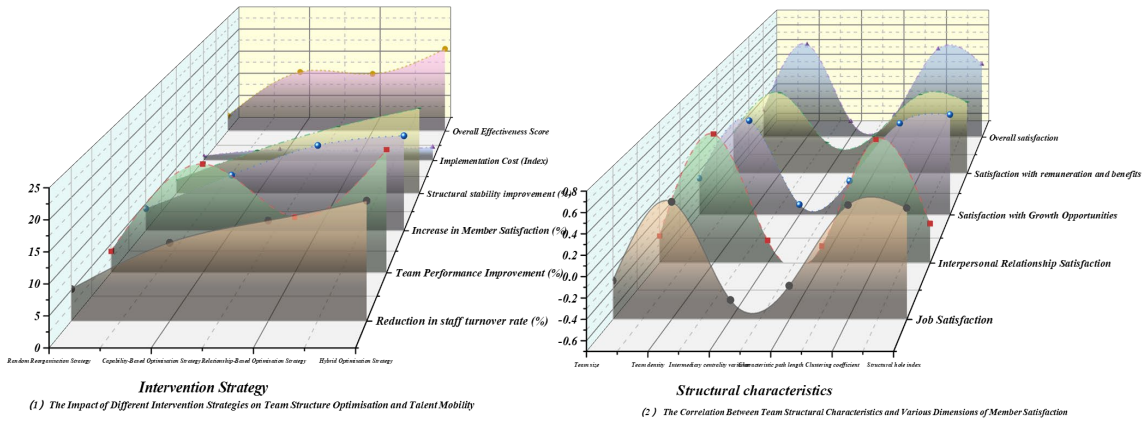


Figure 11. Impact of Different Intervention Strategies on Team Structure Optimisation and Talent Mobility

Figure 12 compares talent mobility patterns across different team development stages. We categorize team development into four phases: forming, storming, norming, and performing, analyzing talent mobility characteristics at each stage. The table reveals that talent attrition rates peak during the forming stage. This may stem from team members being in the initial familiarization phase, where low belongingness makes them more susceptible to leaving due to incompatibility. Attrition rates remain elevated during the storming stage, though the proportion of voluntary departures declines while involuntary departures increase, indicating the team begins weeding out unsuitable members. The turnover rate decreases significantly during the norming stage, with enhanced team member stability. While the execution stage exhibits the lowest turnover rate, the proportion of high-caliber talent leaving increases. This may occur as high-performing members seek new challenges once the team has stabilized. This finding suggests organizations should adopt differentiated talent retention strategies tailored to distinct team development phases.

To facilitate the implementation of the proposed framework in real-world enterprise scenarios, we have designed a three-phase deployment process.

The first stage: Data preparation and graph construction. The enterprise needs to integrate the data from the internal human resource system, project management tools and collaborative office software. After the data cleaning and anonymization, the dynamic organization graph is constructed with employees as nodes and collaboration relationships as edges.

Phase 2: Model Training and Strategy Generation. Train the GAT+MLP model using historical data (e.g., the past 12 months). After training, the model can generate optimization suggestions for each team (e.g., enhancing collaboration among specific employees) and identify core employees at higher risk of leaving, forming retention strategy recommendations.

Phase 3: Intervention Implementation and Feedback Loop. Based on model recommendations, managers implement team adjustments or communicate with high-risk employees. The effectiveness of these interventions (e.g., retention rates of key employees, changes in team performance) is fed back as new data to the model, supporting its continuous learning and optimization. This closed-loop process ensures the model dynamically adapts to organizational changes, providing managers with continuous and precise decision-making support.

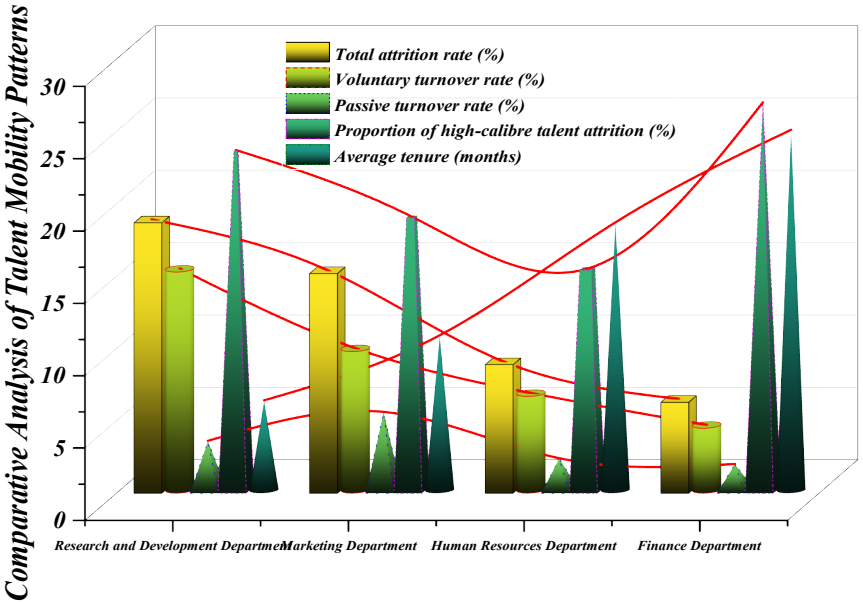


Figure 12. Comparison of talent flow patterns at different stages of team development

6. Conclusion and Outlook

This study constructs a unified, data-driven framework for team structure optimization and talent mobility prediction based on graph neural networks, effectively addressing the complexity and dynamism of modern organizational environments. The method represents over 12,500 employees and more than 56,000 mobility records as a dynamic organizational graph, combining graph attention networks with multilayer perceptrons to jointly model collaboration patterns and mobility behaviors, rather than treating them as isolated problems. This design enables the model to capture local interaction mechanisms and global structural dependencies within large organizations.

Extensive experiments validate the robustness and superiority of the proposed method. In team collaboration prediction, the model achieves an F1 score close to 80% and an AUC higher than 0.81; in talent mobility prediction, it achieves an accuracy of 81.2% and an AUC of 0.864, representing a performance improvement of over 10 percentage points compared to traditional machine learning and recommendation baseline methods. Ablation analysis further demonstrates that combining structural and contextual network features can improve accuracy from 73.5% when using only basic attributes to 81.2% when all features are jointly modeled, highlighting the necessity of integrating multi-source information to achieve reliable predictions. Departmental and time-window experiments confirm that the model has good generalization ability across different business units, and the 12-month time window optimally balances data sufficiency and sensitivity to pattern evolution.

Beyond predictive performance, the collaborative simulation platform also reveals how team structure influences talent mobility. Graph-based metrics (e.g., team size, clustering coefficient, and structural holes) show clear and interpretable correlations with staff turnover rates, while multi-agent simulations demonstrate that a hybrid optimization strategy combining competency matching and relationship quality can reduce staff turnover by 19.5% and improve team performance by 21.4%. Analysis of different team development stages further indicates that differentiated talent retention strategies are needed from the formation to the execution phase, providing actionable guidance for dynamic team management.

Overall, the main innovation of this work lies in combining micro-level graph neural network (GNN)-based predictions with macro-level multi-agent simulations, thereby constructing an integrated process from organizational data to management strategies. The proposed framework not only improves the accuracy and interpretability of team structure and talent mobility analysis but also provides a practical decision support tool for enhancing organizational resilience and management efficiency. Future work could extend this framework across organizational ecosystems and integrate real-time data streams for online adaptive optimization.

References

[1] Hu X, Yu W, Xiao Q. A socially inspired optimization algorithm for multi-extremal function optimization: team innovation algorithm. Soft Computing. 2025; 29 (21-22): 1-15.

[2] Shi X, Jiang A. An optimisation strategy for the organisational structure of human resource management based on social networks. International Journal of Networking and Virtual Organisations. 2025; 32(1-4): 182-202.

[3] Han F, Hu S, Liu S. Research on Organizational Structure Optimization Based on the Theory of Isomers. Economic Society and Humanities. 2024; 1(10): 60-65.

[4] Sistig M H, Sinhuber P, Rogge M, et al. Optimizing Fleet Structure for Autonomous Electric Buses: A Route-Based Analysis in Aachen, Germany. Sustainability. 2024; 16(10): 4093-4117.

[5] Skoko I ,Lušić Z ,Varela S Z , et al. Optimization Model for Selection of the Offshore Fleet Structure. Journal of Marine Science and Engineering. 2024; 12(2): 0263.

[6] Parisa A, Amilcar A, Sandra M. An Optimization Model for Structuring a Car-Sharing Fleet Considering Traffic Congestion Intensity. Journal of Advanced Transportation. 2023; 2023:9283130.

[7] Zhihong J, Xiaohan W, Jiaqing S, et al. Optimization of fleet structure and investment evaluation – the cargo owner's fleet perspective. Maritime Business Review. 2022;7(3): 239-254.

[8] Anna Z, Yuriy L, Anton R. Determination of the structure for the road construction machinery fleet on the basis of fractional

linear optimization. Transportation Research Procedia. 2022;6327-40.

[9] Amphenol ICC and TactoTek Team to Create Automotive Grade In-Mold Connector for Smart Surfaces; Amphenol ICC to extend MicroSpace product line with connectors optimized for mass produced in-mold structural electronics (IMSE) parts.M2 Presswire,2021.

[10] Kherbachi S, Yang Q, Khan Z S. A structured approach to measuring and optimizing the organizational architecture in global product development projects. Concurrent Engineering. 2020;28(3):161-174.

[11] PolozovA A, RolisV A, MikhryakovV S, et al. Team structure optimization using a personal rating calculation in team sports. AIP Conference Proceedings. 2019; 2116(1) :430012.

[12] Terayama K, Yamashita T, Oguchi T, et al. Fine-grained optimization method for crystal structure prediction. npj Computational Materials. 2018;4(1):1-8.

[13] Irmgard L. Natural Structures and Structural Optimization. Journal of the International Association for Shell and Spatial Structures. 2018; 59(1):68-74.

[14] Hum S, Parlar M, Zhou Y. Measurement and optimization of responsiveness in supply chain networks with queueing structures. European Journal of Operational Research. 2018; 264(1):106-118.

[15] CISMILIANU A ,MUNTEANU E C,ONCESCU I, et al. End-to-end process of hollow spacecraft structures with high frequency and low mass obtained with in-house structural optimization tool and additive manufacturing. INCAS BULLETIN. 2017; 9(3):13-20.

[16] Patents; Patent Application Titled "Team Performance by Refining Team Structure" Published Online (USPTO 20170091693).Politics& Government Week. 2017;6873

[17] Farasat A, Nikolaev G A. Signed social structure optimization for shift assignment in the nurse scheduling problem. Socio-Economic Planning Sciences. 2016; 563-13.

[18] GEA works to optimise its group structure[J].Membrane Technology. 2014; 2014(10):1-1.

[19] POWER. Russian Grids tooptimize management structure in2013.Interfax : Russia & CIS Energy Daily,2013.

[20] Best Practices, Llc; Bio-pharma Internal Communications Groups Evolving for Structure Optimization. Biotech Week. 2013.

[21] Gnecco,G., Sanguineti, et al. Suboptimal solutions to team optimization problems with stochastic information structure. SIAM Journal on Optimization: A Publication of the Society for Industrial and Applied Mathematics. 2012; 22(1):212-243.

[22] M A V, G Y A. Problems of farms' machine and tractor fleet structure optimization under market economy.Tractors and Agricultural Machinery. 2011;78(10):6-7.

[23] SHIRAKAWA K, HASHIURA H, SAITO H, et al. Optimization of Grouping and Team Formation of Students for Class Exercise Based on Analysis of Human Factors by Covariance Structure Analysis. WSEAS Transactions on Advances in Engineering Education. 2008; 5(4):176-186.

[24] Brad S, J R L, Darryl S, et al. A controlled investigation of optimal internal medicine ward team structure at a teaching hospital.PloS one. 2012; 7(4):e35576.

[25] Han Z, Xu F, Li Y, et al. Model predicted human mobility explains COVID-19 transmission in urban space without behavioral data. Scientific reports. 2025;15(1) : 6365.

[26] Hajlasz M, Pei S. Predictability of human mobility during the COVID-19 pandemic in the United States. PNAS nexus. 2024; 3(8):308

[27] Waad A, Na M, Francisco S. Automatic prediction of developers' resolutions for software merge conflicts.The Journal of Systems & Software. 2023; 206

[28] Meghna C, Md M S, J TG , et al. Analysis and Prediction of Human Mobility in the United States during the Early Stages of theCOVID-19 Pandemic using Regularized Linear Models. Transportation research record. 2023; 2677(4):380-395.

[29] F A S. Automation of forecasting the number of personnel as the basis of environmental management, on the example of calculations by environmental controllers of enterprises of the fuel and energy complex of Russia. IOP Conference Series: Earth and Environmental Science. 2021; 808(1): 012069.

[30] Iris L G. "Laws of Fear" in the EU: The Precautionary Principle and Public Health Restrictions to Free Movement of Persons in the Time of COVID-19.European Journal of Risk Regulation. 2021; 14(1):141-164.

[31] Science-Applied Sciences. Incheon National University Researchers Publish New Data on Applied Sciences (Comparative Study on Exponentially Weighted Moving Average Approaches for the Self-Starting Forecasting). Science Letter. 2020;756.

[32] Science-Astronomy Research. Researchers at Australian National University Report New Data on Astronomy Research (Oxygen Loss From Simulated Galaxies and the Metal Flow Main Sequence: Predicting the Dependence On Mass and Environment). Chemicals & Chemistry. 2020; 5113.

[33] Engineering-Naval Architecture and Ocean Engineering; Researchers at Korea Maritime and Ocean University Target Naval Architecture and Ocean Engineering (Motion predictive control for DPS using predicted drifted ship position based on deep learning and replay buffer). Journal of Engineering. 2020;2903.

[34] Environmental Informatics. Investigators at Hunan University Detail Findings in Environmental Informatics (Dct-based Least-squares Predictive Model for Hourly Aqi Fluctuation Forecasting). Computer Technology Journal. 2020;263.

[35] Rafael B, Sen P, Jeffrey S, et al. Predicting dengue outbreaks at neighbourhood level using human mobility in urban areas. Journal of the Royal Society Interface. 2020; 17(171):20200691-20200691.