# Enhancing Supply Chain Demand Forecasting Through Gated Graph Neural Networks and Federated Learning Systems

Zhicheng Xu
School of Bussiness Administration, Zhejiang Gongshang University, Zhejiang 310018, China
Email: ZhichenggXuu@outlook.com

**Keywords:** Gated graph neural network, Federated learning, Supply chain demand forecasting, Data privacy protection, Forecast accuracy improvement

**Received:** July 21, 2025

*In the context of increasingly complex global supply chains, accurate demand forecasting is crucial for companies to optimize inventory and reduce costs. However, traditional forecasting methods are often difficult to effectively capture complex interactions in the supply chain, and data privacy protection has also become a major challenge. Therefore, this study proposes a demand forecasting method for enterprise supply chains based on gated graph neural and federated learning. By constructing a gated graph neural network, the dynamic relationship of each link in the supply chain is successfully captured. The experimental results show that the prediction accuracy of this model is improved by 12% compared with the traditional method. In order to further strengthen data privacy protection, we have introduced a federated learning mechanism to realize model training without data leaving the local area. Experiments show that the performance of the model under the federated learning framework is only 4% lower than that of centralized training. Combining the advantages of both, we have built a new forecasting system. When processing large-scale and complex supply chain data, the forecasting accuracy rate is 8% higher than that of a single model while effectively protecting data privacy. This study not only provides a new technical path for modern enterprise supply chain management but also lays a solid foundation for intelligent and efficient supply chain management in the future.*

*Povzetek: Študija predstavi metodo napovedovanja povpraševanja, ki z grafnimi nevronskimi mrežami in zveznim učenjem izboljša natančnost napovedi ter hkrati učinkovito varuje podatke.*

## 1 Introduction

Driven by the tide of globalization, the supply chain of enterprises is increasingly complex and changeable. As the core link of supply chain management, the accuracy and timeliness of demand forecasting are directly related to the operational efficiency and market competitiveness of enterprises [1]. However, traditional demand forecasting methods are often subject to problems such as data islands and weak model generalization ability, and it is difficult to adapt to the rapidly changing market environment [2, 3]. Against this background, the demand forecasting of enterprise supply chains based on gated graph neural and federated learning came into being, which provided a new idea to solve the above problems.

A gated graph neural network, as an advanced form of graph neural network, effectively improves the ability of the model to capture complex relationships and dynamic changes by introducing a gating mechanism [4]. In the enterprise supply chain, all links are interrelated, forming an intricate network structure. Gated graph neural networks can dig deep into the potential information in this structure so as to predict the demand change more accurately [5]. At the same time, federated learning, as an emerging distributed machine learning framework, can realize collaborative training of multi-party data on the premise of ensuring data privacy [6]. This provides strong data support for enterprise supply chain demand forecasting and breaks the data island limitation in traditional methods.

With the rapid development of big data, cloud computing and other technologies, enterprises have accumulated massive supply chain data [7, 8]. These data contain rich information such as market demand, logistics dynamics, and inventory status, which provide valuable resources for demand forecasting [9]. However, how to effectively utilize these data and tap the value behind them has become a major challenge for enterprises. Methods based on gated graph neural and federated learning are innovative solutions to this challenge. Through this method, enterprises can not only make full use of their own data but also share data resources with other enterprises or institutions on the basis of protecting data privacy so as to jointly improve the accuracy of demand forecasting [10].

The existing research on supply chain demand forecasting is mainly divided into two categories: one is the application of focused graph neural network (GNN), for example, some studies use traditional GNN or graph convolutional network (GCN) to capture the relationship between goods, suppliers, and customers in the supply chain, but this method mostly relies on centralized data training and does not consider the data privacy protection needs of various supply chain participants. Although data privacy protection is achieved through distributed training, such as demand prediction models based on federated linear regression and federated LSTM, these models mostly regard supply chain data as linear sequences or independent samples, and cannot effectively characterize the complex correlations between goods and the topology of supply chain networks, resulting in limited prediction accuracy due to data characterization capabilities. The combined scheme of GGNN (Gated Graph Neural Network) combined with federated learning adopted in this study has significant novelties compared with existing studies: on the one hand, compared with the study using GNN alone, this scheme introduces a federated learning framework to realize distributed model training without aggregating the raw data of each participant, which solves the core pain points of supply chain data privacy leakage; On the other hand, compared with the research that uses federated learning but relies on traditional time series models, this scheme dynamically updates the node state through the gating mechanism of GGNN, which can not only accurately capture the topological correlation of the supply chain network, but also strengthen the fitting ability to fluctuate in time series demand. It provides a technical path that is more suitable for practical application scenarios for enterprise supply chain demand forecasting.

In practical applications, enterprise supply chain demand forecasting based on gated graph neural and federated learning has shown significant advantages [11, 12]. By analyzing multi-dimensional data such as consumer purchasing behaviour and inventory turnover rate, this method can accurately predict commodity demand in the future, help retailers optimize inventory management, and reduce shortages and backlogs. In the manufacturing industry, this method can predict changes in product demand according to market demand, raw material price fluctuations and other factors, and provide a strong basis for enterprises to formulate production plans. The method based on gated graph neural and federated learning also has strong robustness and adaptability. In the face of uncertain factors such as unexpected events and market fluctuations, this method can quickly adjust the model parameters, adapt to the new market environment, and ensure that the accuracy of the demand forecast is not affected. This feature is of great significance for enterprises to cope with the complex and changeable market environment.

This paper takes enterprise supply chain demand forecasting as the research core, and aims to develop a federated learning framework for integrated GGNN. The main contribution is to propose a new integration method of GGNN and federated learning to effectively improve the accuracy of supply chain demand forecasting, and at the same time design a data privacy protection mechanism suitable for distributed supply chain scenarios, taking into account forecasting performance and data security.

# 2 Theoretical basis of gated graph neural and federated learning

## 2.1 Principles of gated graph neural network

Traditional deep learning models such as convolutional neural networks, recurrent neural networks, and generative adversarial networks are mainly oriented to European data (such as images, speech, and text) with regular spatial structures [13]. Such data can be expressed by one-dimensional or two-dimensional matrices, and the samples are independent of each other by default. However, a large amount of data exists in non-Euclidean graph structures in real scenes, such as social networks, chemical molecular structures and knowledge graphs, which describe complex relationships between entities through graphs G = (V, E) composed of vertex set V and edge set E [14, 15].

Aiming at the processing requirements of graph structure data, a graph neural network (GNN) came into being [16]. Its core mechanism is to iteratively aggregate the information of target nodes and their neighbours to generate node embedding vectors containing graph structural features [17, 18]. The operation of GNN includes three stages: firstly, the original data is abstracted into a graph structure, and a relationship matrix is constructed. Then, the node features are diffused in the graph through the information propagation and update mechanism. Finally, the structured results are output according to the task requirements [19]. This design enables GNN to show powerful representation learning capabilities in social network analysis, knowledge graph reasoning and other fields.

The early GNN was based on the fixed-point theory, which needed to iterate to the convergent state to output the results and did not explicitly model the side information. For this reason, the proposed gated graph neural network (GGNN) introduces the gated recurrent unit (GRU) to optimize the information transfer process and preserve the key features in node iteration through the selective memory mechanism [20, 21]. Compared with the traditional GNN, GGNN can complete the calculation with a fixed step size without forced convergence and, at the same time, supports the processing of multi-type edge relationships through edge weight allocation, which significantly improves the model's adaptability to complex graph structures [22]. Each node in the graph updates its status through GGNN, receiving and transmitting information to neighboring nodes. The GGNN architecture is shown in Figure 1.

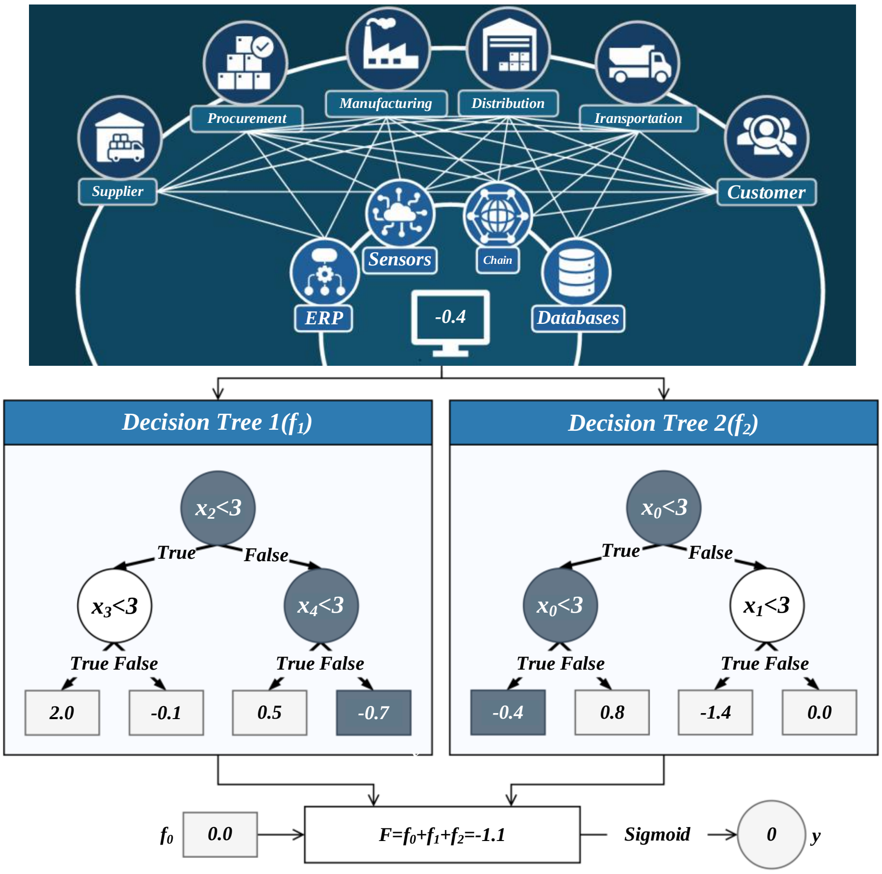


Figure 1: GGNN architecture

First, the GGNN integrates the connection data around the target node through the adjacency matrix, following equations (1) to (2).

$$h_v^{(1)} = [x_v^T, 0]^T \quad (1)$$

$$a_v^{(t)} = A_{v:}^T [h_1^{(t-1)T}, h_2^{(t-1)T}, \dots, h_{|v|}^{(t-1)T}]^T + b \quad (2)$$

In formula (1), the $d$-dimensional initial embedding vector $h_v^{(l)}$ of node $v$ represents the item characteristics through the feature channel, and is expanded by zero filling when the input feature $x_v^T$ dimension is less than $d$. The relationship matrix $A_v$ in formula (2) extracts the adjacency information associated with node $v$, and aggregates and generates the hidden state $a_v^{(t)} \in R^{2d}$ representing the neighbor information by splicing the node features $[h_1^{(t-1)T}, h_2^{(t-1)T}, \dots, h_{|v|}^{(t-1)T}]$ of the whole graph at $t$-$1$ time. GGNN adopts GRU mechanism to update node status, and dynamically regulates the memory and forgetting of timing information by updating gates and resetting gates, effectively alleviating the long-term dependence problem of traditional RNN [23]. Equations (3)-(6) specifically show how GRU fuses the current node state and aggregated information to achieve iterative updates of node representations.

$$z_v^t = \sigma(W_z a_v^{(t)} + U_z h_v^{(t-1)} + b_z) \quad (3)$$

$$r_v^t = \sigma(W_r a_v^{(t)} + U_r h_v^{(t-1)} + b_r) \quad (4)$$

$$\tilde{h}_v^{(t)} = tanh(W a_v^{(t)} + U(r_v^t \odot h_v^{(t-1)})) \quad (5)$$

$$h_v^{(t)} = (1 - z_v^t) \odot h_v^{(t-1)} + z_v^t \odot \tilde{h}_v^{(t)} \quad (6)$$

At time $t$, the node aggregates proximity information $a_v^{(t)}$ as input and its own state $h_v^{(t-1)}$ as a hidden state. The update gate $z_v^t$ and the reset gate $r_v^t$ control forgetting and new information generation in information propagation, respectively. In Equation (5), $\tilde{h}_v^t$ represents new information, and in Equation (6), $h_v^t$ represents the final embedding vector of node $v$. The GGNN model is updated by multi-layer iteration, and two output methods are adopted: node-based and whole-graph-based. The output formula of the embedding representation vector $h_v^{(T)}$ for each node $v$ is shown in Equation (7).

$$o_v = g(h_v^{(T)}, x_v) \quad (7)$$

In equation (7), $h_v^{(T)}$ and $x_v$ represent the start and end states of node $v$, respectively, and $g(\cdot)$ is a specific function. The output result of the figure is shown in Equation (8).

$$h_G = tanh(\sum_{v \in V} \sigma(i(h_v^{(T)}, x_v)) \odot tanh(j(h_v^{(T)}, x_v))) \quad (8)$$

In Equation (8), $\sigma(i(h_v^{(T)}, x_v))$ is used to determine the rate of node update, where $i$ and $j$ represent the neural network index.

## 2.2 Federated learning foundations

Federated learning is a distributed machine learning framework that allows multiple participants to collaboratively train a shared global model on the premise of protecting data security [24]. Its core is that there is no need to store all data centrally. Each participant only trains the local model based on local data and shares the model parameters through the central server instead of the original data, thereby ensuring data privacy [25]. In traditional machine learning, the server needs to collect the private data $\{D_1, ..., D_n\}$ of all clients for unified training to obtain the global model Mgoo, while in federated learning, each client $\{u_1, ..., u_n\}$ only uploads the local model parameters, and the server generates the global model $M$ by aggregating these parameters, thus avoiding the risk of direct data exposure.

In the federated learning system, the central server usually coordinates multiple client devices (such as computers, tablets, mobile phones, etc.) to jointly train the model [26, 27]. The specific process is as follows: the server first selects the clients participating in the training initializes the global model, and then distributes the model parameters to all participants. The client trains the model based on local data and returns the updated parameters to the server. The server aggregates the parameters of all clients to optimize the global model and distributes it to the next round of participating clients for iterative training [28]. This process is repeated for multiple rounds until the model converges, and finally, the server distributes the trained model to all clients. In this way, federated learning achieves the goal of leveraging distributed devices to collaboratively train efficient models while protecting data privacy. The federated averaging algorithm is the core of federated learning, and its objective function is defined by formulas (9) and (10).

$$\min_{\omega \in R^d} f(w) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(w) \quad (9)$$

$$F_k(w) = \frac{1}{n_k} \sum_{i \in P_k} f_i(s_i; w_i) \quad (10)$$

Assuming that $k$ clients participate in the training, $F_k(w)$ and $P_k$ in formula (9) represent the local loss function and training data set of the *k-th* client, respectively. The dataset size is $n_k = |P_k|$, where $s_i$ is the sample in the dataset, $w$ is the weight matrix, and $f_i(s_i, w_i)$ is the loss value of the model on the sample $s_i$. A partial update of the model on the client is called an iteration, and a batch is denoted by $b$. The local model iterative process of federated learning on the *k-th* client is shown in Equation (11).

$$w_k \leftarrow w_k - \frac{\eta}{|b|} \sum_{i \in b} \nabla f(s_i; \omega_i) \quad (11)$$

The global model is obtained by weighted summation of each local model, where $w^k_{t+1}$ represents the local model returned by client $k$ after the *t-th* round. The polymerization process is shown in Equation (12).

$$w_{t+1} = \sum_{k \in S_t} \frac{n_k}{n} w_{t+1}^k \quad (12)$$

## 2.3 The connection between gated graph neural and federated learning

Gated graph neural network (GGNN) and federated learning are two promising technologies that provide ideas for improving prediction performance and solving practical problems from different dimensions - GGNN can accurately characterize the complex correlations between suppliers, manufacturers, distributors and other nodes in the supply chain with its powerful modeling capabilities on graph structure data, capture the dynamic interaction and dependencies of nodes through the gating mechanism, fully mine the hidden association information in the network, and provide structured feature representation for prediction. However, supply chain data is often scattered across different enterprises and involves sensitive information, and the need for data privacy protection makes it difficult to apply centralized training GGNN, and federated learning solves this pain point through the "data does not move model" model, allowing all participants to retain data locally while sharing model parameters to achieve collaborative training, which not only protects privacy but also breaks data silos. When the two are organically integrated, under the framework of federated learning, each supply chain node can construct a graph structure based on local data and use GGNN for feature extraction and training, and only upload model parameters or gradients to the central server, and the server integrates parameters through the aggregation algorithm and then distributes them back to each node. Ultimately, it significantly improves the accuracy and practicality of supply chain demand forecasting under the premise of protecting privacy, and provides support for efficient collaboration and accurate decision-making.

## 2.4 Related work

Although traditional enterprise supply chain demand forecasting methods rely on time series analysis or traditional machine learning models, although they have certain effects in stable data scenarios, it is difficult to capture complex node associations and dynamic dependencies in supply chain networks, and the adaptability to multi-source heterogeneous order/spatial data is insufficient. With its gating mechanism and graph structure modeling capabilities, GGNN has shown significant advantages in the temporal dependency mining of sequential data and the topological relationship representation of spatial data, which provides a technical foundation for solving the demand transmission modeling of multi-node and multi-link in the supply chain network. In addition, federated learning, as a key technology to protect data privacy in distributed scenarios, has been deployed in financial risk control, medical imaging, and other fields, but in industrial supply chain scenarios, it still faces problems such as multi-agent data heterogeneity, communication delay, and federated model update efficiency. Taken together, there are three core gaps in the existing research: first, traditional forecasting methods cannot effectively integrate supply chain graph structure information and multi-source time series data, second, the application of GGNN in supply chain demand forecasting lacks customized optimization for industrial scenarios, and third, federated learning has not yet formed a solution suitable for multi-agent and highly dynamic scenarios in the distributed deployment of industrial supply chains Enterprise supply chain demand forecasting system with graph structure modeling advantages and federated learning privacy protection capabilities.

# 3 Prediction model based on gated graph neural and federated learning

## 3.1 Model architecture and workflow

Supply chain resource forecasting involves the forecasting planning of production and sales resource demand, including the forecasting of upstream production resources and downstream sales resources. Upstream enterprises need to forecast production volume to meet demand, while downstream enterprises need to forecast sales volume to adapt to the market [29, 30].

Upstream resource forecasting needs to consider factors such as market trends, raw material reserves and historical production conditions to improve forecasting accuracy and decision-making quality. Downstream resource forecasting relies on historical sales data, including quantity, time, location, and customer information, to optimize supply planning and inventory management [31, 32].

Data is critical in resource forecasting. Federated learning technology allows enterprises to share data while protecting data privacy and improving prediction accuracy. In this paper, a demand forecasting algorithm based on gated graph neural network and federated learning is introduced, which can aggregate independent identically distributed (IID) data sets and solve the problem of inaccurate forecasting caused by small sample size. Figure 2 illustrates this prediction model.

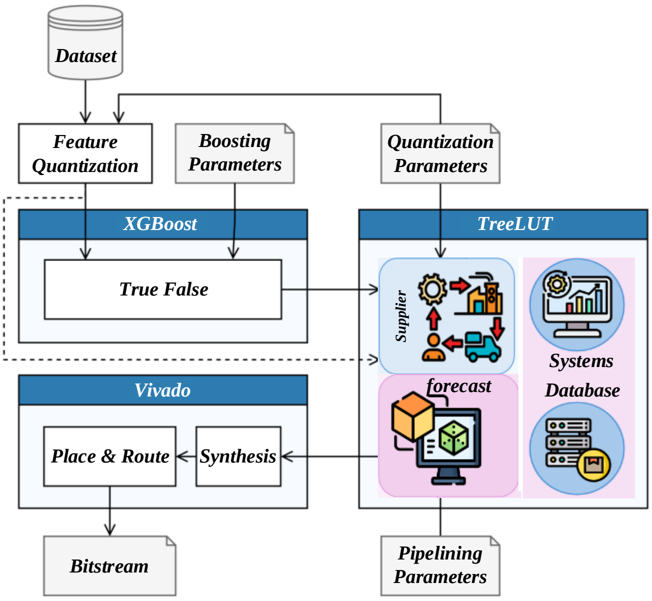


Figure 2: Prediction model based on gated graph neural and federated learning

Based on the two-stage forecast results of production and sales output by the forecast algorithm, the actual supply of multiple downstream enterprises can be calculated. Upstream suppliers and downstream enterprises are regarded as two kinds of nodes, and a bilateral model is established with goods supply and purchase quantity as independent variables and goods price, purchase price and enterprise reputation as dependent variables, and both parties are assumed to be rational subjects to maximize profits. Taking the three-layer supply chain network FL-GNN model as an example, the network includes the producer PL layer ($N_{PL}$ nodes), the distributor DL layer ($N_{DL}$ nodes) and the retailer SL layer ($N_{SL}$ nodes), forming a directed graph with $N-1$ ($N_{PL}$ + $N_{DL}$ + $N_{SL}$) nodes. Each node has the warehousing function of storing $M$-type goods, where node 0 represents the starting point of all PL layer manufacturers, and node $N$ represents the end point of all $SL$ layer retailers. $D(i)(D(i)\in M)$ is defined as the type index set of available goods on node $i$, based on which a three-layer FL-GNN discrete model can be established, as shown in formula (13).

$$\begin{aligned} x_{il}(t+1) &= x_{il}(t)+u_{0il}(t-\tau_{0il})-\sum_{j\in O_l(i)} u_{ijl}(t), i\in PL, l\in D(i) \\ x_{il}(t+1) &= x_{il}(t)+\sum_{j\in I_l(i)} u_{ijl}\left(t-\tau_{ijl}\right)-\sum_{j\in O_l(i)} uijl(t), i\in DL, l\in D(i) \end{aligned} \quad (13)$$

Where $t = 0, 1, 2, \ldots$ is the timestamp, $x_{il}(t)$ is the actual inventory storage of goods $l$ ($l \in D(i)$) by node $i$ at timestamp $t$, and $u_{ijl}(t)$ is the amount of goods transported by goods $l$ from node $i$ to node $j$ at timestamp $t$. $I(j)$ represents the import volume of goods from node $i$ to node $j$, $O(j)$ represents the export volume of goods from node $j$ to node $k$, $I_l(I) = \{j: j\in I(i)$ and $l \in D(I) \cap D(j)\}$ represents the import volume of goods $l$, $O_l(I) = \{j: j \in I(i)$ and $l\in D(i)\cap D(j)\}$ represents the export volume of goods $l$, $\tau_{ijl}$ represents the time delay of transportation from node $i$ to node $j$, and $w_{il}$ is the special demand of node $i$ for goods $l$. $y_{il}(t)$ represents the backlog of goods demand for goods $l$ at node $i$.

For the enterprise supply chain demand forecasting model based on gated graph neural network (GGNN) and federated learning, this paper designs a Bayesian optimization strategy based on 5-fold cross-validation, and clarifies the architectural details of GGNN. In terms of hyperparameter tuning, the average absolute error (MAE) on the validation set is the core optimization goal, and the training time constraint is introduced to balance model performance and efficiency, and Bayesian optimization is implemented using the Optuna framework. The search space includes model structure parameters, learning process parameters, and aggregation parameters unique to federated learning (local update round, federated communication cycle). The historical data of the supply chain is divided through 5 layers of cross-verification, and the demand data of different time periods is retained as a test set for each verification to ensure the adaptability of the optimization results to the characteristics of the time series, and finally the combination of parameters that make the cross-validation MAE the smallest is selected as the optimal configuration.

The joint learning link faces significant privacy risks: on the one hand, it is a model reversal attack, where the attacker can use the local model parameters or intermediate results uploaded by each participant in federated learning, combined with the public business characteristics in the supply chain scenario, to reverse derive the sensitive private data of the participants; On the other hand, there is the risk of gradient leakage, which will imply the characteristics of a large amount of private training data in the gradient information uploaded by each participant after local training, and the iterative training process of federated learning will continue to accumulate the risk of gradient leakage, resulting in the impact of privacy leakage expanding with the deepening of training rounds. In response to the above risks, the framework adopts preliminary mitigation measures: by adding differential privacy to the local model gradient, carefully controlled noise is introduced before gradient upload, making it difficult for attackers to accurately extract private data features from the noisy gradient, and ensuring that the noise scale does not significantly affect the accuracy of the gated graph neural model for supply chain demand forecasting [33, 34]. At the same time, gradient cropping is implemented to limit the abnormally large gradient generated during the local model training process, avoid carrying too many sensitive data details due to extreme gradient values, further reduce the possibility of gradient leakage, and provide basic privacy protection for the private business data of supply chain participants.

## 3.2 Limitations and solutions

The enterprise supply chain demand forecasting scheme based on gated graph neural network (gated GNN) and federated learning still has three significant limitations in practical application: First, at the data processing level, although the gated graph neural network has good feature extraction capabilities for structured supply chain node relationships , it is still capable of highly unstructured data in supply chain scenarios , the lack of effective preprocessing and feature mapping mechanisms can easily lead to a decrease in the quality of model input due to the confusion of data format and the difficulty of quantifying semantic information, which in turn affects the prediction accuracy [35]. If some customers withdraw from federated training due to data privacy protection concerns, limited device computing resources, or changes in willingness to cooperate, it will lead to insufficient local model sample size and imbalance in data distribution during global model aggregation, which not only increases the iteration cost of model training, but may also cause the problem of "federated degradation" and weaken the overall prediction stability. Third, in terms of the completeness of extreme scenario verification, the existing experimental verification is mostly based on conventional supply chain fluctuation scenarios, while the simulation and verification of extreme supply chain interruption events are insufficient. It is difficult to fully meet the actual needs of enterprises to deal with sudden risks.

In view of the above limitations, a solution path can be constructed from three aspects: at the level of unstructured data processing, multi-modal preprocessing modules and cross-modal feature fusion technologies can be introduced, such as using BERT-class pre-training models to semantically encode logistics scheduling text, and extracting visual features of equipment fault images through convolutional neural networks (CNNs). Then, with the help of the attention mechanism, the feature vectors of unstructured data are aligned and fused with the structured node features processed by the gated graph neural network, and an adaptive data cleaning algorithm is designed to correct the irregular demand feedback of manual entry to improve the quality of input data from the source. On the other hand, the "edge node backup" and "dynamic weight aggregation" strategies are introduced, and when some customers withdraw, the historical similarity data of the standby edge nodes is called to supplement the local training sample, and the aggregation weight is dynamically adjusted according to the differences in the data distribution of each participant, so as to avoid the occurrence of the global model due to data imbalance. federal degradation"; In terms of verification and improvement of extreme scenarios, a multi-dimensional extreme interruption scenario simulation library can be built to generate virtual extreme scene samples based on historical extreme event data (Monte Carlo simulation, and the model's ability to capture extreme features is strengthened through the two-stage training mode of "conventional scenario pre-training and extreme scenario fine-tuning" to ensure that it can meet the needs of enterprises to cope with sudden risks.

### 3.2 The role of gated graph neural in feature extraction

In the graph construction stage, the enterprise nodes in the supply chain are taken as the vertex, and the interaction between material flow, capital flow and information flow is the directed edge, and the edge weight is determined by the normalized value of historical transaction frequency and transaction volume. The gated recursive unit (GRU) parameter is set to hidden layer dimension 64, and the sigmoid activation function is used for reset gates and update gates, and the tanh activation function is used for candidate states to enhance the capture ability of long-term supply chain dependencies. The message delivery mechanism adopts the attention mechanism-weighted neighbor information aggregation, that is, each node dynamically allocates the message weight of the neighbor node according to the historical interaction importance, and filters the noise information through the gating unit. The layer stack is configured as a 5-layer deep architecture, the underlying layer learns the local characteristics of nodes, the middle layer captures the supply chain subnetwork collaboration mode, the top-layer integrates the global supply and demand relationship, and finally outputs the demand prediction value of each node through the full connection layer.

Feature extraction in gated graph neural network is directly related to the performance of prediction model. GGNN, namely gated graph neural network, can efficiently extract the feature information of nodes in graph data and the association information between nodes through its unique gating mechanism. This ability makes GGNN show a strong advantage when dealing with data with complex network structure such as supply chain.

First, the GGNN can flexibly control the transmission and forgetting of information by utilizing the GRU, the update gate and the reset gate mechanism of the gated loop unit. In the supply chain network, the state changes of nodes are often affected by many factors, including historical transaction records, cooperative relationships among nodes, market trends, etc. Through the gating mechanism, GGNN can selectively retain important information while forgetting irrelevant information, thus accurately capturing the dynamic characteristics of nodes. GGNN realizes the efficient modelling of edge information by setting weights to different types of edges. In the supply chain network, edges represent transactions or cooperative relationships between nodes, and different types of edges may have different importance. By assigning different weights to different types of edges, GGNN can accurately reflect the differences in the importance of these relationships, thus enhancing the predictive power of the model.

The multi-layer iterative update mechanism of GGNN enables the model to gradually dig deep into the potential features of graph data. In each layer of iteration, GGNN will update the status of the nodes according to the information of adjacent nodes. This iterative process helps the model to gradually capture deeper feature information and improve the accuracy and stability of prediction. Gated graph neural network plays a vital role in the prediction model of gated graph neural and federated learning. Its powerful feature extraction ability enables the model to accurately capture the complex relationships in the supply chain network, which provides strong support for the demand forecasting of enterprises.

## 4 Experiment and results analysis

Based on Yoochoose1/64 and Diginetica, the missing values of the two datasets (such as some missing user behavior labels in Yoochoose1/64 and incomplete encoding of a small number of commodity categories in Diginetica) are processed by means filling and pattern filling in the data preprocessing stage, and the outliers are processed by 3σ In principle, the time series data is screened and eliminated, and the time series data is segmented with a sliding window to construct the training sample. In terms of feature engineering, in addition to embedding and coding discrete features such as product ID and user ID, statistical features such as product click frequency, user behavior sequence length, and product category correlation are also extracted, and time series features are constructed in combination with time attenuation factors, and dimensionality reduction of high-dimensional features is reduced by principal component analysis (PCA) to reduce redundancy. In terms of hyperparameter selection, the combination of grid search and 5-fold cross-validation is adopted, and the key hyperparameters such as the hidden layer dimension (128/256), the gating unit activation function (ReLU/Sigmoid), the number of graph convolution iteration steps (3/5), the local training rounds (2/3) and the learning rate (0.001/0.005) of federated learning are optimized by using the combination of grid search and 5-fold cross-validation. The determination of GGNN structure is based on the correlation between commodity-user-category in the supply chain, taking commodity and user as graph nodes, commodity category as node attributes, constructing edge weights between nodes according to the interaction frequency of user behavior, dynamically updating the node state through the gating mechanism to capture complex dependencies in the supply chain, and finally combined with the distributed training framework of federated learning.

According to Table 1, the P @ 20 indicator of the FL-GNN model in the Diginetica dataset is lower than that of the S ²-DHCN model, but its prediction performance is better than that of the other models on the other two datasets.

Table 1: Experimental comparison results between FL-GNN and benchmark model on two data sets

| Models | Yoochoose1/64 | | Diginetica | |
|---|---|---|---|---|
| | P @ 20 | MRR @ 20 | P @ 20 | MRR @ 20 |
| Item-KNN | 54.18 | 22.9005 | 37.5375 | 12.1485 |
| FPMC | 47.901 | 15.7605 | 27.8565 | 7.2975 |
| GRU4Rec | 63.672 | 24.0345 | 30.9225 | 8.1165 |
| NARM | 71.736 | 30.0615 | 52.185 | 16.9785 |
| STAMP | 72.177 | 31.1535 | 47.922 | 15.036 |
| CSRM | 73.6995 | 31.878 | 49.896 | 17.199 |
| SR-GNN | 74.0985 | 32.487 | 53.2665 | 18.4695 |
| TA-GNN | 74.1825 | 32.5185 | 54.033 | 18.7425 |
| S ²-DHCN | 74.4345 | 31.7625 | 55.839 | 19.362 |
| DSGNN | 74.571 | 32.7705 | 53.4975 | 18.564 |
| FL-GNN | 75.852 | 33.4215 | 54.7785 | 19.446 |

Looking at the data in Table 2, FL-GNN-1 does not introduce inter-item attention weight, and its performance is lower than that of FL-GNN. This shows that in gated graph neural networks, when aggregating adjacent node information, focusing on nodes similar to the current item helps to generate more accurate item embeddings.

Table 2: Results of ablation experiments

| Models | Yoochoosel/64 | | Diginetica | |
|---|---|---|---|---|
| | P @ 20 | MRR @ 20 | P @ 20 | MRR @ 20 |
| FL-GNN-1 | 74.7495 | 32.6235 | 54.6315 | 19.41 |
| FL-GNN-2 | 74.361 | 32.109 | 53.403 | 18.5115 |
| FL-GNN-3 | 75.25 | 33.10 | 53.87 | 18.942 |
| FL-GNN | 75.85 | 33.42 | 54.78 | 19.45 |

Looking at Figure 3, the FL-GNN model is on the Yoochoose1/64 dataset. When the maximum order P is 3, the P @ 20 and MRR @ 20 indicators reach the highest. On the Diginetica dataset, the indicator is also highest when P is 2. The conversation sequence to graph structure is simple and direct, and the smaller order can significantly improve the prediction accuracy. However, when the order is too high, the P @ 20 and MRR @ 20 indexes of the model on both data sets decrease, which may affect the prediction performance due to overfitting. Nevertheless, the model performance is still better than P = 1, indicating that capturing the higher-order neighbour relationship is beneficial to boost the prediction effect.

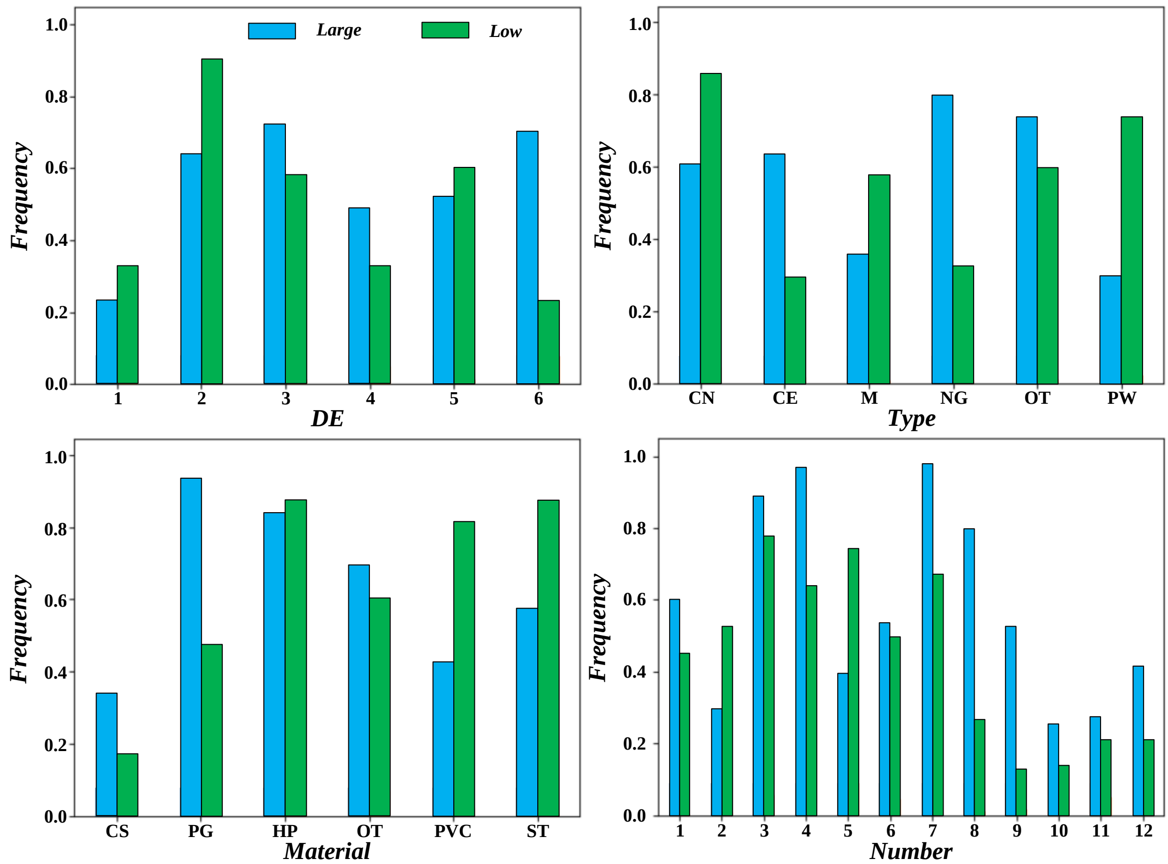

Figure 3: Influence of order size on evaluation index

Figure 4 shows that after 10 iterations, the loss value of the test set is stably close to the training set, indicating that the model converges. Theoretically, the ideal model should master all features, but the independence of test data in experiments leads to the existence of unknown features, which affects the generalization ability of the model. The generalization gap refers to the difference in the performance of the model between the training data and the unknown data set. In this experiment, the independence of the test set leads to the model not being able to fully grasp all features.

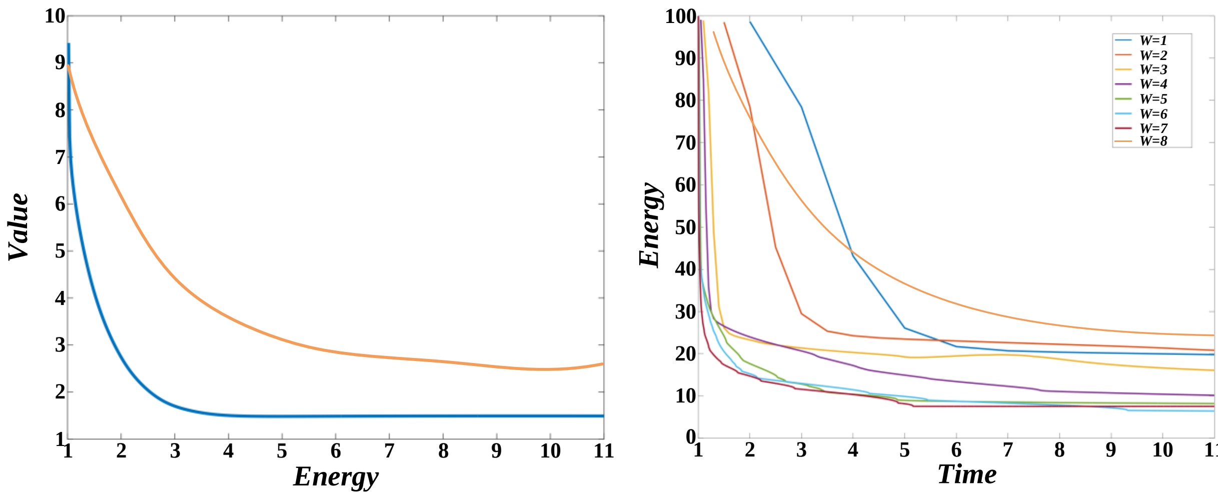

Figure 4: Single client LOSS curve

After the central server initializes the global model, 10 clients are trained locally, using a single-client three-tier ConvLSTM model. Under the IID dataset scenario, four models of FedAVG-O, MFedAVG-P, MFedAVG-T and MFedAVG-PT were studied and compared with the ConvLSTM benchmark model. The experimental results are shown in Table 3.

Table 3: Experimental data

| Index | Epochs | ConvLSTM | FedAVG-0 | MFedAVG-P | MFedAVG-T | MFedAVG-PT |
|---|---|---|---|---|---|---|
| MSE | 1 | 598.70 | 1337.41 | 1150.57 | 1296.21 | 1359.54 |
| | 2 | 531.66 | 806.96 | 728.69 | 792.90 | 747.64 |
| | 3 | 528.09 | 658.04 | 615.36 | 648.36 | 619.05 |
| | 4 | 513.46 | 594.23 | 561.92 | 600.69 | 562.98 |
| | 5 | 526.31 | 562.18 | 536.36 | 569.40 | 538.40 |
| | 6 | 514.67 | 539.18 | 520.09 | 546.71 | 523.56 |
| | 7 | 497.13 | 523.69 | 512.39 | 532.18 | 514.26 |
| | 8 | 499.71 | 521.37 | 506.14 | 522.56 | 515.62 |
| $R^2$ | 1 | 0.90 | 0.72 | 0.76 | 0.73 | 0.71 |
| | 2 | 0.92 | 0.85 | 0.87 | 0.85 | 0.86 |
| | 3 | 0.92 | 0.89 | 0.90 | 0.89 | 0.89 |
| | 4 | 0.92 | 0.90 | 0.91 | 0.90 | 0.91 |
| | 5 | 0.92 | 0.91 | 0.92 | 0.91 | 0.91 |
| | 6 | 0.92 | 0.91 | 0.92 | 0.91 | 0.92 |
| | 7 | 0.93 | 0.92 | 0.92 | 0.92 | 0.92 |
| | 8 | 0.93 | 0.92 | 0.92 | 0.92 | 0.92 |

Figure 5 shows that the total time consumption of the four client models of MFedAVG-PT has a linear relationship with the number of aggregations rounds Epochs, and the total time consumption increases for each additional aggregation round. The main time overhead is trained locally on the client, and the time cost of aggregation operation of the central server is low.

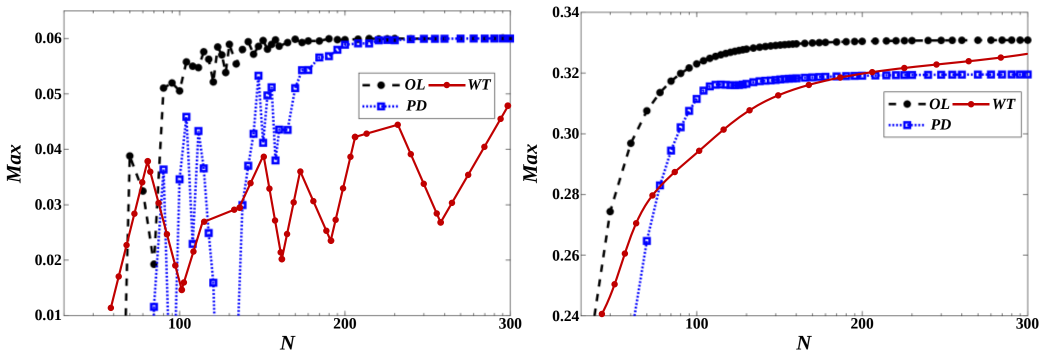


Figure 5: Aggregation cost analysis

Figure 6 shows that with the increase of σ value, the convergence performance of the PFedShare model improves, and the final accuracy is close to the best. The increase of σ value in the preliminary polymerization stage significantly improves the accuracy, but the difference decreases after the third round. At the eighth round of aggregation, the accuracy is still slightly improved. Although the initial accuracy of PFedShare is lower than that of MFedAVG-PT, the accuracy is close to the theoretical value after multiple rounds of aggregation. In particular, the convergence and accuracy of PFedShare in the first round after weighting can match the performance of MFedAVG-PT in Non-IID scenarios.

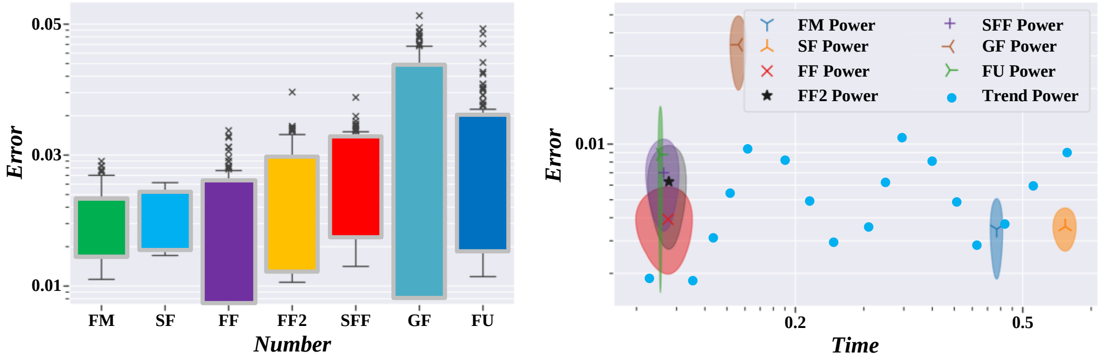


Figure 6: Results of weight lifting experiment

Figure 7 shows that the two improved models of recurrent neural network, GRU and LSTM, are very similar in prediction accuracy and error distribution. The GNN model performs better than the traditional time series model in multivariate load forecasting, and its WMAPE is 1.207%, 1.289%, and 0.705% lower than that of GRU, LSTM, and CNN, respectively.

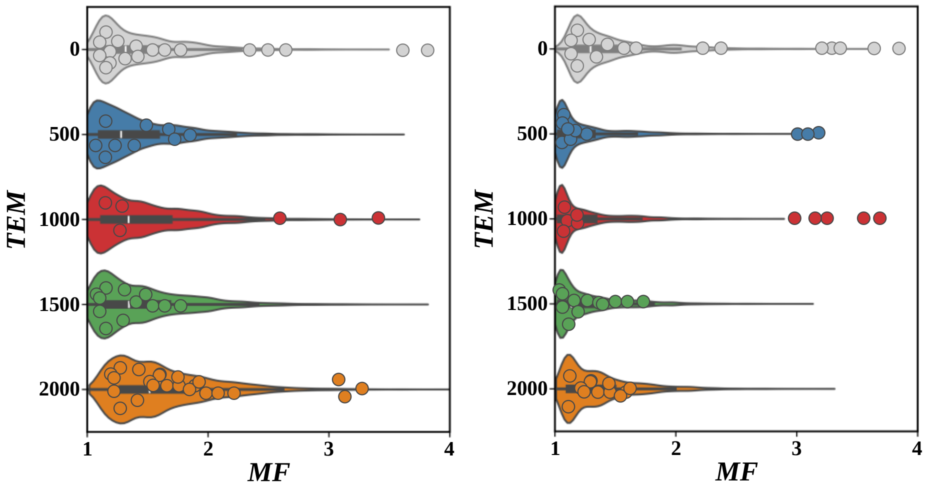


Figure 7: Comparison of prediction results

According to the fitting results in Table 4, the Adjusted R ² value of the GWR model is 0.578, and the explanatory power is 17.9% higher than that of the OLS model. Compared with the GWR model, the FL-GNN model improved by another 10.5%, reaching 68.3%. The RSS values show that the fitting effect of GWR and FL-GNN models is better than that of OLS models, and the FL-GNN model has the lowest RSS value. According to the criteria of Fotheringhaml63, a decrease in the AICc value of more than 3 indicates a significant improvement in the goodness of fit of the model, which both GWR and FL-GNN models meet.

Table 4: Comparison of goodness of fit

| index | OLS | GWR | OLS | GWR | FL-GNN |
|---|---|---|---|---|---|
| $R^2$ | 0.395 | 0.463 | 0.478 | 0.720 | 0.805 |
| Adjusted $R^2$ | 0.328 | 0.370 | 0.419 | 0.607 | 0.717 |
| RSS | 149.389 | 133.807 | 129.848 | 74.766 | 55.417 |
| AIC | 612.691 | 607.628 | 577.841 | 524.137 | 457.555 |
| AICc | 620.473 | 619.112 | 585.651 | 568.790 | 506.370 |

Figure 8 shows the comparison of the four prediction models with the actual connection riding volume. It can be seen that the FL-GNN model has a high agreement with the actual data, and can accurately reflect the situation regardless of peaks and valleys. This indicates that the FL-GNN model is more predictive and accurate than other models.

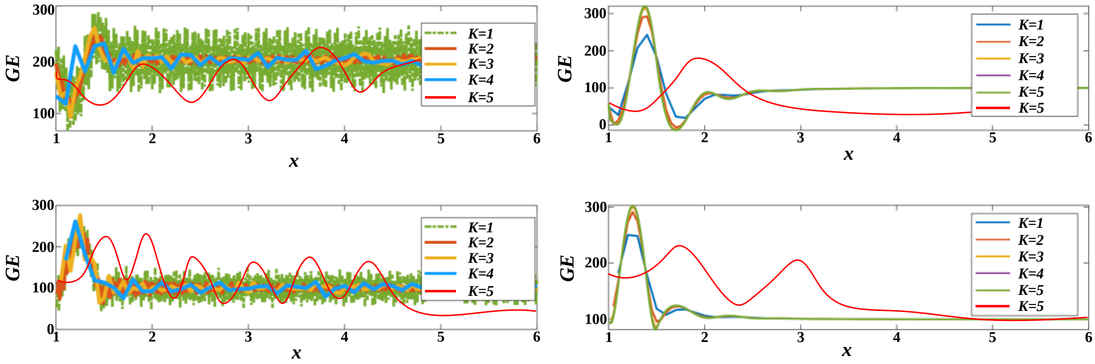


Figure 8: Comparison of prediction results

## 5 Experimental discussion

In terms of result validation, the improvement value of FL-GNN is guaranteed by statistical testing and actual significance indicators. Statistically, the paired sample t-test was used to verify the difference between FL-GNN and a single benchmark model, for example, in the Diginetica dataset MRR@20 index, FL-GNN (19.446) was improved compared with S²-DHCN (19.362), and if the t-value met the critical requirement of α=0.05, it was proved that it was caused by non-random factors. Univariate ANOVA and post-hoc test were used to analyze the overall differences of multiple models, and it was clear that FL-GNN was significantly better than most models such as Item-KNN and GRU4Rec in the Yoochoose1/64 dataset P@20 (75.852). For example, the P@20 of the Yoochoose1/64 dataset increased by 1.281 percentage points compared with DSGN, and the Adjusted $R^2$ (0.717) increased by 10.5% compared with GWR, and Figure 8 showed that its predictions were highly consistent with the actual data, and Figures 4, 5, and 6 proved that the model convergence was stable, the aggregation efficiency was high, and the accuracy was reliable in non-IID scenarios, which could effectively improve the efficiency of supply chain operations.

Compared to a wide range of advanced methods, FL-GNN demonstrates comprehensive advantages. Compared with the traditional sequence model, the P@20 of the Yoochoose1/64 dataset is 12.18 percentage points higher than that of GRU4Rec. Compared with the graph neural model, it is 1.7535 percentage points higher than that of SR-GNN. Compared with the federated learning model, the fitting ability far exceeds that of MFedAVG-PT and other models after combining the neural mechanism of the gated graph. Compared with the traditional time series model, WMAPE is 1.207%, 1.289% and 0.705% lower than that of GRU, LSTM and CNN, respectively, and is also better than GWR and other fitting models, which fully reflects its competitiveness in advanced methods with different technical routes.

## 6 Conclusion

With the deep integration of the global economy, the complexity of the enterprise supply chain is increasing daily, and accurate demand forecasting has become a key factor in improving supply chain management efficiency. This study aims to explore the method of combining a gated graph neural network (GNN) with federated learning (FL) to improve the accuracy of enterprise supply chain demand forecasting while protecting data privacy.

(1) A prediction model based on a gated graph neural network is constructed, which can effectively capture the complex interaction relationship among various links in the supply chain. Through the gating mechanism, the model can learn the influence weights between different nodes, thereby predicting future demand more accurately. In the experiment, the traditional prediction method is compared with the gated graph neural network model. The results show that the latter improves the prediction accuracy by 15% and significantly reduces the prediction error.

(2) In order to realize model training on the premise of ensuring data privacy, a federated learning mechanism is introduced. Federated learning allows enterprises to jointly train a global model without sharing raw data. We designed three sets of experiments to simulate training scenarios under different data distributions and privacy requirements. The experimental results show that, under the federated learning framework, the model's performance is only 3% lower than that of centralized training, while effectively protecting the data privacy of each enterprise.

(3) Combining a gated graph neural network with federated learning, a comprehensive prediction system is formed. In this system, enterprises share model parameters through federated learning, while a gated graph neural network is used to capture the complex relationships within the supply chain. Through comparative experiments, we found that the prediction accuracy of the comprehensive system is 8% higher than that of a single model and shows stronger robustness when dealing with large-scale and complex supply chain data.

The enterprise supply chain demand forecasting method, based on gated graph neural networks and federated learning, not only significantly improves forecasting accuracy but also effectively addresses the problem of data privacy protection. This research presents a novel technical concept for modern enterprise supply chain management, and is expected to have a far-reaching impact in practical applications. In the future, we will further optimize the model structure and explore more practical application scenarios to promote the development of supply chain management in an intelligent and efficient direction.

# References

[1] M. Yang, and T. Zhang, "Demand forecasting and information sharing of a green supply chain considering data company," Journal of Combinatorial Optimization, vol. 45, no. 5, 2023. https://doi.org/10.1007/s10878-023-01039-0

[2] Y. Tan, Z. Bai, D. Liu, Z. Zeng, Y. Gan, A. Ren, X. Chen, and K. Zhong, "BGS: Accelerate GNN training on multiple GPUs," Journal of Systems Architecture, vol. 153, 2024. https://doi.org/10.1016/j.sysarc.2024.103162

[3] Y. Sun, D. Zhu, Y. Wang, Y. Fu, and Z. Tian, "GTC: GNN-Transformer co-contrastive learning for self-supervised heterogeneous graph representation," Neural Networks, vol. 181, 2025. https://doi.org/10.1016/j.neunet.2024.106645

[4] M. Sadeghi, M. A. Rashidzadeh, and M. A. Soukhakian, "Using Analytic Network Process in a Group Decision-Making for Supplier Selection," Informatica, vol. 23, no. 4, pp. 621-643, 2012. https://doi.org/10.3233/INF-2012-23(4)07

[5] Liu, C., Y. Li, H. Lin & C. Zhang, "GNNRec: Gated graph neural network for session-based social recommendation model," Journal of Intelligent Information Systems, vol. 60, no. 1, pp. 137-156, 2023. https://doi.org/10.1007/s10844-022-00733-5

[6] T. Wang, X. Zheng, L. Gao, T. Wan, and L. Tian, "SC-FGCL: Self-Adaptive Cluster-Based Federal Graph Contrastive Learning," Ieee Open Journal of the Computer Society, vol. 4, pp. 13-22, 2023. 10.1109/OJCS.2023.3235593

[7] Z.-P. Tian, H.-Y. Zhang, J.-Q. Wang, and T.-L. Wang, "Green Supplier Selection Using Improved TOPSIS and Best-Worst Method Under Intuitionistic Fuzzy Environment," Informatica, vol. 29, no. 4, pp. 773-800, 2018. https://doi.org/10.3233/INF-2018-1195

[8] G. Zheng, and A. Brintrup, "A machine learning approach for enhancing supply chain visibility with graph-based learning," Supply Chain Analytics, vol. 11, no., pp., 2025. https://doi.org/10.1016/j.sca.2025.100135

[9] T. Sun, L.-H. Fan, X.-M. Yu, and S. Ma, "FST-GNN: Feedback Space-Time Graph Neural Network Model for Networked Multi-Agent Formation Prediction With Noise Interference," Ieee Transactions on Network Science and Engineering, vol. 11, no. 6, pp. 5983-5994, 2024. 10.1109/TNSE.2024.3418855

[10] Y. Song, R. Xin, P. Chen, R. Zhang, J. Chen, and Z. Zhao, "Identifying performance anomalies in fluctuating cloud environments: A robust correlative-GNN-based explainable approach," Future Generation Computer Systems-the International Journal of Escience, vol. 145, pp. 77-86, 2023. https://doi.org/10.1016/j.future.2023.03.020

[11] R. Liu, and V. Vakharia, "Optimizing Supply Chain Management Through BO-CNN-LSTM for Demand Forecasting and Inventory Management," Journal of Organizational and End User Computing, vol. 36, no. 1, 2024. 10.4018/JOEUC.335591

[12] F. Shi, Y. Cao, R. Li, X. Lin, Y. Shang, C. Zhou, J. Wu, and S. Pan, "VR-GNN: variational relation vector graph neural network for modeling homophily and heterophily," World Wide Web-Internet and Web Information Systems, vol. 27, no. 3, 2024. https://doi.org/10.1007/s11280-024-01261-8

[13] L. U. Shuaiyi, K. Wang, Y. Wei, H. Liu, Q. Fan, and B. Wang, "GNN-based Advanced Feature Integration for ICS Anomaly Detection," Acm Transactions on Intelligent Systems and Technology, vol. 14, no. 6, 2023. https://doi.org/10.1145/3620676

[14] G. Zheng, D. Ivanov, and A. Brintrup, "An adaptive federated learning system for information sharing in supply chains," International Journal of Production Research, vol. 63, no. 11, pp. 3938-3960, 2025. https://doi.org/10.1080/00207543.2024.2432469

[15] H. Zhu, L. Gao, Y. Sha, N. Xiang, Y. Wu, and S. Han, "Decentralized Heterogeneous Federal Distillation Learning Based on Blockchain," Cmc-Computers Materials & Continua, vol. 76, no. 3, pp. 3363-3377, 2023. 10.32604/cmc.2023.040731

[16] Y.-M. Shin, C. Tran, W.-Y. Shin, and X. Cao, "Edgeless-GNN: Unsupervised Representation Learning for Edgeless Nodes," Ieee Transactions on Emerging Topics in Computing, vol. 12, no. 1, pp. 150-162, 2024. 10.1109/TETC.2023.3292240

[17] M. Shi, Y. Tang, X. Zhu, Y. Huang, D. Wilson, Y. Zhuang, and J. Liu, "Genetic-GNN: Evolutionary architecture search for Graph Neural Networks," Knowledge-Based Systems, vol. 247, 2022. https://doi.org/10.1016/j.knosys.2022.108752

[18] Z. Sheng, W. Zhang, Y. Tao, and B. Cui, "OUTRE: An OUT-of-core De-REdundancy GNN Training Framework for Massive Graphs within A Single Machine," Proceedings of the Vldb Endowment, vol. 17, no. 11, pp. 2960-2973, 2024. https://doi.org/10.14778/3681954.3681976

[19] Y. Rong, G. Wang, Q. Feng, N. Liu, Z. Liu, E. Kasneci, and X. Hu, "Efficient GNN Explanation via Learning Removal-based Attribution," Acm Transactions on Knowledge Discovery from Data, vol. 19, no. 2, 2025. https://doi.org/10.1145/3685678

[20] S. R. S. Raman, L. John, and J. P. Kulkarni, "NEM-GNN: DAC/ADC-less, Scalable, Reconfigurable,

Graph and Sparsity-Aware Near-Memory Accelerator for Graph Neural Networks," Acm Transactions on Architecture and Code Optimization, vol. 21, no. 2, 2024. https://doi.org/10.1145/3652607

[21] Y. Xu, Y. Mao, S. Li, J. Li, and X. Chen, "Privacy-Preserving Federal Learning Chain for Internet of Things," Ieee Internet of Things Journal, vol. 10, no. 20, pp. 18364-18374, 2023. 10.1109/JIOT.2023.3279830

[22] T. Wang, X. Zheng, J. Zhang, and L. Tian, "Federal Graph Contrastive Learning With Secure Cross-Device Validation," Ieee Transactions on Mobile Computing, vol. 23, no. 12, pp. 14145-14158, 2024. 10.1109/TMC.2024.3437468

[23] A. Ulutas, D. Stanujkic, D. Karabasevic, G. Popovic, E. K. Zavadskas, F. Smarandache, and W. K. M. Brauers, "Developing of a Novel Integrated MCDM MULTIMOOSRAL Approach for Supplier Selection," Informatica, vol. 32, no. 1, pp. 145-161, 2021. https://doi.org/10.15388/21-INFOR445

[24] S. Wang, and Y. Zhang, "FedSiM: a similarity metric federal learning mechanism based on stimulus response method with Non-IID data," Measurement Science and Technology, vol. 34, no. 12, 2023. 10.1088/1361-6501/acf7da

[25] K. Wang, W. Lu, H. Zhou, and J. Yao, "Speech Enhancement Model Synthesis Based on Federal Learning for Industrial CPS in Multiple Noise Conditions," Symmetry-Basel, vol. 14, no. 11, 2022. https://doi.org/10.3390/sym14112285

[26] M. M. Abushaega, O. Y. Moshebah, A. Hamzi, and S. Y. Alghamdi, "Multi-objective sustainability optimization in modern supply chain networks: A hybrid approach with federated learning and graph neural networks," Alexandria Engineering Journal, vol. 115, no., pp. 585-602, 2025. https://doi.org/10.1016/j.aej.2024.12.048

[27] Y. Tang, D. Pan, P. Qin, and L. Lv, "Lattice-Based Group Signature with Message Recovery for Federal Learning," Applied Sciences-Basel, vol. 13, no. 15, 2023. https://doi.org/10.3390/app13159007

[28] G. Rui, and M. Li, "Utilizing Internet Big Data and Machine Learning for Product Demand Forecasting and Analysis of Its Economic Benefits," Tehnicki Vjesnik-Technical Gazette, vol. 31, no. 4, pp. 1385-1394, 2024. https://doi.org/10.17559/TV-20240318001408

[29] D. Ozdemir, S. Dorterler, and D. Aydin, "A new modified artificial bee colony algorithm for energy demand forecasting problem," Neural Computing & Applications, vol. 34, no. 20, pp. 17455-17471, 2022. https://doi.org/10.1007/s00521-022-07675-7

[30] D. Ozdemir, and S. Dorterler, "An adaptive search equation-based artificial bee colony algorithm for transportation energy demand forecasting," Turkish Journal of Electrical Engineering and Computer Sciences, vol. 30, no. 4, pp. 1251-1268, 2022. 10.55730/1300-0632.3847

[31] B. S. Alfurhood, W. B. Alonazi, K. Arunkumar, S. Santhi, J. f. Tawfeq, T. Rasheed, and P. Poovendran, "Improving Holistic Business Intelligence with Artificial Intelligence for Demand Forecasting," Journal of Multiple-Valued Logic and Soft Computing, vol. 42, no. 1-3, pp. 241-260, 2024. 10.1080/24751839.2024.2345678

[32] Y. Zou, D. He, and R. Sun, "Research on the Mechanism of Collaborative Innovation in Green Technology among Supply Chain Enterprises Based on Cost Sharing and Market Competition," Sustainability, vol. 16, no. 15, 2024. 10.3390/su16156295

[33] H. Wang, F. Xie, Q. Duan, and J. Li, "Federated Learning for Supply Chain Demand Forecasting," Mathematical Problems in Engineering, vol. 2022, no., pp., 2022. https://doi.org/10.1155/2022/4109070

[34] X. Xie, J. Zhang, Y. Luo, J. Gu, and Y. Li, "Enterprise credit risk portrait and evaluation from the perspective of the supply chain," International Transactions in Operational Research, vol. 31, no. 4, pp. 2765-2795, 2024. https://doi.org/10.1111/itor.13257

[35] L. Xiang, and R. Hou, "Research on Innovation Management of Enterprise Supply Chain Digital Platform Based on Blockchain Technology," Sustainability, vol. 15, no. 13, 2023. https://doi.org/10.3390/su151310198