Journal of Network Intelligence ©2024 ISSN 2414-8105
Taiwan Ubiquitous Information Volume **9**, Number **3**, August **2024**

Trade and Economic Forecasting Based on Genetic Optimisation Graph Neural Networks

Li-Jia Wang*

Xuchang Electrical Vocational College, Henan 461000, P. R. China
ghwlj8500@163.com

Guang-Min Liu

Investors Group Securities Inc., Hwy Pointe-Claire, QC H9R 0A5, Canada
gh3189973@163.com

*Corresponding author: Li-Jia Wang
Received November 8, 2023, revised February 15, 2024, accepted April 23, 2024.

Abstract. *Through trade and economic forecasting, the demand and change trend of commodities can be predicted in advance, which can help e-commerce enterprises to optimise the supply chain management and improve the coordination of logistics and procurement. However, the composition of heterogeneous data information of various types of commodities is more complex, so how to identify commodity features quickly and accurately becomes the key to this problem. Heterogeneous graph neural network can extract and integrate feature information from heterogeneous data of multiple sources, so as to improve the trade data prediction accuracy. Therefore, this work proposes a trade and economic prediction method based on genetic optimisation graph neural network. Firstly, a novel knowledge graph construction method is given to be designed and a knowledge graph of trade goods entry and exit data based on inclusion relationship is constructed. Secondly, for the problem of low quality and low efficiency of multi-source heterogeneous data retrieval, based on the knowledge graph, genetic algorithm is used to search for the optimal structure of the heterogeneous graph neural network model, including the number of nodes, the number of layers, and the size of convolution kernel. By defining the fitness function, the genetic algorithm can search the optimal network structure among the candidate structures. Finally, taking a large port as an example, the genetic optimisation graph neural network model is used to predict the port logistics demand, and the results are compared with those predicted by other similar prediction models. The experimental results show that the Genetically Optimised Graph Neural Network model is more accurate and stable than other prediction models.*
Keywords: e-commerce; trade economy; knowledge graph; graph neural network; genetic algorithm

---

1. **Introduction.** E-commerce trade and economic forecasting can improve the operational efficiency and user experience of e-commerce platforms, and provide more accurate transaction forecasts and personalised recommendations for enterprises and consumers [1, 2]. At the same time, it can promote the development of e-commerce, enhance the status of e-commerce in the economy, and promote the development of network economy.

Through e-commerce trade and economic forecasting, resource planning and optimisation can be better carried out. For example, predicting the demand for goods can adjust the production plan to avoid excessive or insufficient production and make effective use of production resources; predicting the market feedback can optimise the marketing strategy and avoid waste of resources. Accurate trade and economic forecasting results can provide enterprises with important decision-making basis, such as the planning of marketing activities, product pricing, sales target setting, etc [3, 4]. At the same time, it can also help enterprises to discover market trends and competitors' movements in advance, optimise business models and seize market opportunities. Through e-commerce trade and economic forecasting, it can predict the demand for commodities and change trends in advance, help enterprises optimise supply chain management, rationally allocate resources, reduce inventory pressure, and improve the cooperation of human resources, logistics and procurement [5, 6].

Trade and economic forecasting [7, 8] refers to the prediction of future trade trends and economic development trends through the analysis and modelling of economic variables, trade data and other relevant factors. Related studies mainly include the application of time series analysis, regression models, artificial neural network models and other methods. In addition, it also involves research on the prediction of macroeconomic indicators and the impact of trade policies on the economy [9, 10, 11]. However, the composition of heterogeneous data information of various types of commodities is more complex, so how to quickly and accurately identify the characteristics of commodities has become the key.

Graph Neural Networks (GNNs) can extract and integrate feature information from heterogeneous data from multiple sources [12, 13], thus improving trade data prediction accuracy. E-commerce trade and economic prediction based on GNNs can improve prediction accuracy, optimise supply chain management, improve resource utilisation efficiency, and promote the development of network economy. This has important application value and practical significance for both enterprises and social economic development.

1.1. **Related Work.** Commodity information recognition [14] refers to the extraction of various attributes and features of commodities by analysing and recognising the textual content of commodity-related information. Related research mainly includes the application of techniques such as text mining [15], natural language processing and machine learning [16].

The related research covers the recognition of commodity names, sentiment analysis of description texts, and feature extraction of commodity reviews [17]. Commodity information recognition is very important for trade and economic forecasting. Commodity information recognition can help researchers to obtain detailed information about different commodities in the market, including brand, model, specification, and origin. This information is very important for forecasting trade activities and helps in predicting the market development.

In recent years, with the rapid development of big data and artificial intelligence technology, data-driven methods have been widely used in commodity information identification. By mining and analysing a large amount of trade data and commodity information, more accurate and comprehensive commodity characteristics and forecasting results can be extracted. Theofilatos et al. [18] used machine learning algorithms to predict exchange rate fluctuations. The researchers used a variety of machine learning models, including random forests and support vector machines, to analyse and predict exchange rate data. The results show that machine learning algorithms can effectively predict exchange rate movements and are better in predictive performance than traditional economic models. Song et al. [19] proposed a method based on text analytics that can automatically extract key information, such as brand, price, and specification, from product descriptions. It is proved through experiments that the method has high accuracy and efficiency in extracting commodity information. This literature provides an automated method for e-commerce platforms to process product information, which can improve the accuracy and user experience of product search. Xu et al. [20] proposed a deep learning-based smart retail architecture for IoT, which uses Convolutional Neural Networks (CNNs) for feature extraction and recognition of product images. Sun et al. [21] proposed a Bag of Visual Words (BoVW)-based approach to extract features from merchandise images and combine them with a deep learning model for classification. Zou et al. [22] proposed a Deep Convolutional Neural Network (DCNN)-based model that was trained on a large-scale e-commerce product database and was used to recognise merchandise information. The results show that the deep convolutional neural network can learn the high-level features of the product images and has a strong representation ability.

1.2. **Motivation and contribution.** Compared with other deep neural network models, GNNs models have more obvious advantages in large-scale heterogeneous commodity information recognition. Large-scale heterogeneous commodity information can usually be naturally represented as a graph structure, where different types of nodes and edges represent different types of commodities and relationships between commodities. GNNs are specifically designed to process graph data, and are able to efficiently capture the structural and relational information between nodes, which gives them a natural advantage in processing graph data. Therefore, to address the problem of low quality and low efficiency in retrieving heterogeneous data from multiple sources, this work proposes the use of GNNs models for commodity information recognition to achieve the goal of trade and economic forecasting.

The main innovations and contributions of this work include:

(1) A novel knowledge graph construction method is designed and a data knowledge graph of traded commodities based on inclusion relationships is constructed so that multiple factors such as commodity codes, commodity attributes, transaction history, etc., can be considered comprehensively. GNNs can cope well with the problem of cross-modal information fusion. By connecting different types of nodes and edges into the same graph, GNNs can fuse different modalities together and achieve cross-modal information transfer and integration.

(2) In order to further improve the performance of the GNNs model, a genetic algorithm (GA) is used to search for the optimal structure of the heterogeneous GNNs model, including the number of nodes, the number of layers, and the size of convolution kernel. By defining the fitness function, the GA algorithm can search the optimal network structure among the candidate structures. Heterogeneous commodity information may involve multiple modalities, such as text, image, video, etc.

2. **Introduction to the rationale.**

2.1. **Principles of GA algorithm.** Genetic algorithm is an optimisation algorithm based on the principles of natural selection and genetics [23]. It finds the optimal solution by modelling the evolutionary process in nature.

The GA algorithm is an optimisation algorithm inspired by natural selection and genetic mechanisms. It simulates the genetic and evolutionary mechanisms in the process of biological evolution, and gradually optimises the solutions in the search space through operations such as inheritance, mutation, selection and crossover of individuals in the population [24]. The basic principle of GA algorithm is to simulate the evolutionary process in nature, and find the optimal solution through continuous evolution and selection. The main steps of the genetic algorithm include:

(1) Initialising the population: a set of initial solutions, called the population, is randomly generated.

(2) Assessing fitness: each individual is assessed for fitness, i.e., the quality of its solution is calculated. The fitness function f(xi) is used to assess the quality of each individual's solution, which is usually the value of the objective function.

(3) Selection operation: some individuals are selected as parents based on fitness and used to generate the next generation. Selection operation is used to select some individuals as parents based on fitness for generating the next generation. The formula for selection operation is as follows:

$$P(x_i) = \frac{f(x_i)}{\sum_{j=1}^{N} f(x_j)} \tag{1}$$

where N denotes the population size, x_i denotes the chromosome of the i-th individual, $f(x_i)$ denotes the fitness of the i-th individual, and $P(x_i)$ denotes the fitness of the i-th individual.

(4) Crossover operation [25]: crossover operation is performed on the parent to produce new individuals. Crossover operation is used to perform crossover operation on the parent to produce new individuals. The formula for crossover operation is as follows:

$$x_{i,j} = \begin{cases} x_{p,j} & \text{if rand}() < P_c \\ x_{i,j} & \text{otherwise} \end{cases} \tag{2}$$

where P_c denotes the crossover probability and $x_{i,j}$ denotes the j-th gene of the i-th individual.

(5) Mutation operations: Mutation operations are used to perform mutations on new individuals to produce more diversity. Mutation operation is used to perform mutation operation on new individuals to produce more diversity. The formula for the mutation operation is as follows:

$$x_{i,j} = \begin{cases} 1 - x_{i,j} & \text{if rand}() < P_m \\ x_{i,j} & \text{otherwise} \end{cases} \tag{3}$$

where P_m denotes the probability of variation.

(6) Assessing adaptation: assessing adaptation in new individuals.

(7) Selection operation: selecting some individuals as parents of the next generation based on fitness. The sorting operation is used to sort the individuals according to the fitness for the selection operation. The calculation formula for the sorting operation is as follows:

$$r_i = \text{rank}(f(x_i)) \tag{4}$$

where r_i denotes the ranking of the i-th individual.

The cumulative value of selection probability is used to select some individuals as parents of the next generation based on fitness. The formula for calculating the cumulative value of selection probability is as follows:

$$s_i = \sum_{j=1}^{i} P(x_j) \tag{5}$$

where s_i denotes the cumulative value of the selection probability of the i-th individual.

(8) Repeat Steps 4-7 until the stop condition is met.

2.2. **Introduction to heterogeneous GNNs.** Graph Neural Networks (GNNs) are a class of machine learning models for processing graph data [26, 27], which extend traditional neural network approaches in order to be able to efficiently deal with unstructured graph data, such as social networks, knowledge graphs, biological networks, etc. GNNs are designed to capture the relationships between nodes in graph data, enabling the representation of a node to better take into account the information of its neighbouring nodes. This gives GNNs a wide range of applications in tasks such as node classification, link prediction, graph classification and recommender systems.

Graph Convolutional Network (GCN) is a key member of GNNs.The core idea of GCN is to consider the representation of each node as a weighted average of the representations of its neighbouring nodes. This weighted average allows a node to propagate information over the graph while taking into account its directly connected neighbouring nodes.The basic idea of a GCN is to capture the local structural information of a node by performing a convolution operation on the nodes in the graph. In GCN, the new features of each node are a function of its own features and the features of its neighbours.

The basic principle of GCN involves the normalisation of the adjacency matrix [27].

$$\hat{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \tag{6}$$

where A is the adjacency matrix of the graph and D is the diagonal matrix whose diagonal element D_{ii} is the degree of node i.

The feature propagation is shown below:

$$H^{(l+1)} = \sigma(\hat{A} H^{(l)} W^{(l)}) \tag{7}$$

where $H^{(l)}$ is the node feature matrix of layer l, $W^{(l)}$ is the weight matrix of layer l, and σ is the activation function, e.g., the ReLU function.

The output layer in GCN is calculated as shown below:

$$Z = \sigma(\hat{A} H^{(L)} W^{(L)}) \tag{8}$$

where Z is the node feature matrix of the output layer, H^L is the node feature matrix of the last layer and W^L is the weight matrix of the last layer.

The loss function is calculated as shown below:

$$L = -\sum_{i=1}^{n} \sum_{j=1}^{F} Y_{ij} \log(Z_{ij}) \tag{9}$$

Where L is the loss function, n is the number of nodes, F is the number of categories, Y is the true labelling matrix and Z is the predicted labelling matrix.

The GCN backpropagation is calculated as shown below:

$$\frac{\partial L}{\partial W^{(l)}} = \frac{\partial L}{\partial H^{(l+1)}} \frac{\partial H^{(l+1)}}{\partial W^{(l)}} \tag{10}$$

where $W^{(l)}$ is the weight matrix of layer l and $H^{(l+1)}$ is the node feature matrix of layer $l+1$.

The weight update is achieved by gradient calculation in the following manner:

$$W^{(l)} = W^{(l)} - \alpha \frac{\partial L}{\partial W^{(l)}} \tag{11}$$

where α is the learning rate.

The advantages of GCNs in multi-source heterogeneous information feature recognition are that they can effectively fuse and process information from different data sources, improve feature recognition performance, and are adaptable and flexible for multi-source information integration and feature learning tasks in different domains. Multi-source heterogeneous information usually includes different types of nodes and edges, as well as different features and relationships. GCNs can effectively fuse this information and use graph structure to integrate information from different data sources, making the integrated information more comprehensive. For example, heterogeneous information usually includes different types of relationships, not just node features.GCNs can adaptively handle different types of relationships to better capture the correlation between different information sources.

3. Heterogeneous commodity information knowledge map construction.

3.1. **Data collection.** According to the new version of import and export commodity information coding standards, the international trade commodity information data set containing large-scale data was established.

The web pages were fetched using the requests framework in a Python 3.8 environment and the unstructured data from the web pages of the General Administration of Customs and the Ministry of Commerce were converted into HTML language. In this paper, the crawling has been done using selenium framework, which is initially an automated testing tool that uses the browser kernel to actually access the web pages. For the obtained HTML pages, BeautifulSoup can be used to obtain the specified tags, after obtaining the text description of the product and the customs code where the tag, use the select method to extract the content of the tag that is to obtain the required text data.

The original dataset was taken from multiple publicly available websites, and because the data came from multiple websites, the data were structured differently and of varying quality. There exists a lot of low quality or unsuitable data for use in this work. Therefore, firstly, conventional cleansing is performed to extract the more worthwhile parts from the entire semi-structured text using rules. Then, the segmentation of the declared elements is performed by means of the markers present in the raw data in order to facilitate the construction of the knowledge graph afterwards. Finally, data enhancement was performed for some of the missing data and very few commodity codes.

3.2. **Entity extraction and relationship extraction in knowledge graphs.** Entities in a knowledge graph are usually described by words or phrases in the text. The knowledge graph constructed in this paper is a heterogeneous graph [29] that contains three types of entities: commodities, declaration elements, and keywords.

Commodity entities, as commodity nodes, are descriptions of the overall information of the commodity; declared element entities, as declared element nodes, are descriptions of the characteristics of a particular aspect of the commodity; keyword entities, as keyword nodes, are keywords extracted from the declared elements. Take the commodity description "Adult Toothpaste | Oral Care Toothpaste | Whitening" as an example, three types of entities can be extracted from it, the commodity entity is "Adult Toothpaste | Oral Care Toothpaste | Whitening"; the declared element entities include:" Adult toothpaste", "oral care toothpaste", "whitening"; keyword entities include: "oral", " toothpaste", "adult", "whitening".

The knowledge graph of trade commodities constructed in this paper contains three types of entities: commodities, declaration elements, and keywords. The relationship between commodities and declaration elements is the specification of declaration elements. The relationship between declared elements and keywords is a single relationship, which is defined in this paper as a "containment relationship", i.e., declared elements contain their corresponding keywords.

3.3. **Knowledge graph modelling and storage.** In this paper, we use RDF (Resource Description Framework) as the data model of the knowledge graph of traded commodities, which is a knowledge representation model proposed by W3C to describe and express the content and structure of the resources on the Internet. The basic data unit of RDF is a ternary, which can be expressed as ¡subject, predicate, object¿, and each ternary represents a fact or a piece of knowledge. Each ternary represents a fact or a piece of knowledge. There is a "containment" relationship between the declarative element entity and the keyword entity. The visualisation of these ternary relationships mapped in the knowledge graph is shown in Figure 1.

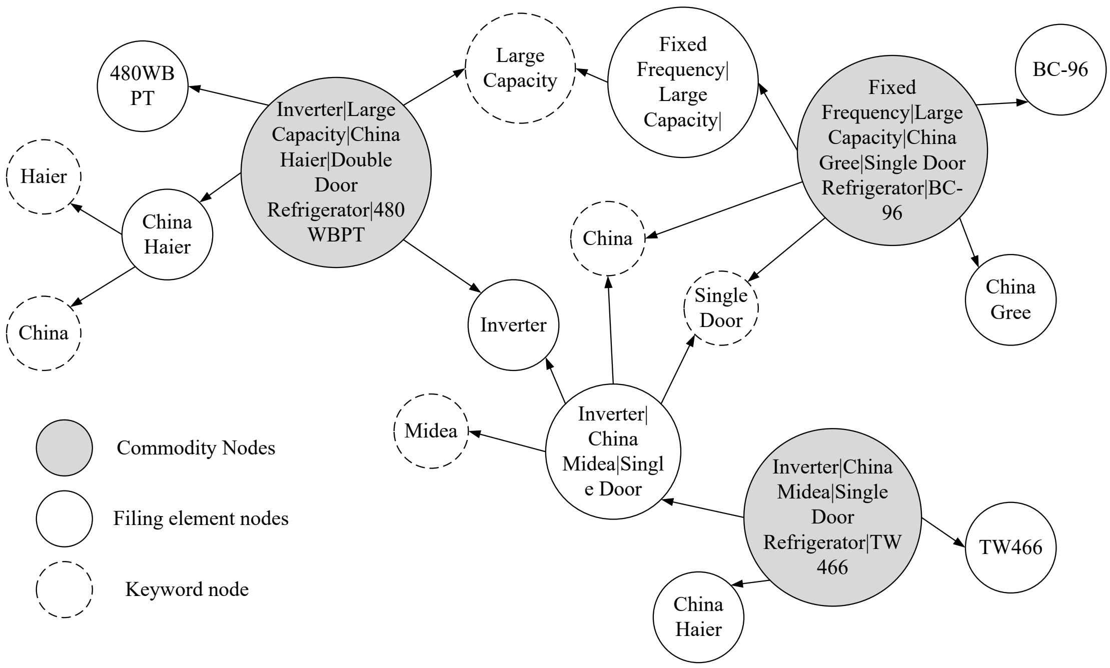

Figure 1. Example of subgraph from knowledge graph

In order to manage the knowledge graph data efficiently, it is necessary to organise these data properly on the storage medium. A text file is used to store the knowledge graph of traded goods. The knowledge graph can be seen as a collection of triples. Three text files are ultimately used to store the Trade Goods Knowledge Graph. The first file is the entity ID file, which stores the textual representation of each entity, the corresponding entity ID and entity type. The second file is the relationship ID file, storing the textual representation of each relationship, the corresponding relationship ID. the third file is the ternary file, which is stored in the form of each line representing a set of ¡header entity ID, relationship ID, tail entity ID¿. The number of nodes and the number of triples for each knowledge graph are shown in Table 1.

Table 1. Knowledge graph information

Data set	Number of nodes	Number of ternary groups
Summer Men's Clothing Data	23484	64271
Data on agricultural equipment	76338	108686
Data on kitchen appliances	44022	59242
Data on domestic vehicles	13074	21140
Data on medical and health equipment	24742	38257

4. **Trade and economic forecasting based on heterogeneous GA-GCN models.**

4.1. **GA-GCN model.** The advantage of GCN is that it can effectively capture local information between nodes with shared weight parameters [30], so it also performs well on large graphs with a large number of nodes and edges. However, it also has some limitations, such as difficulty in handling heterogeneous and global information in the graph.

In order to further improve the performance of the GNNs model, a Genetic Algorithm (GA) is used to search for the optimal structure of the heterogeneous GNNs model, including the number of nodes, the number of layers, and the size of convolutional kernel. Firstly, a set of candidate network structures need to be initialised. Each network structure can be encoded as a chromosome, where the genes represent various parameters of the network, such as the number of nodes, the number of layers, the convolution kernel size, and so on. This process can be represented as:

$$C = \{c_1, c_2, ..., c_n\} \tag{12}$$

where C is the set of chromosomes, c_i is the i-th chromosome, and n is the number of chromosomes.

Define a fitness function to evaluate the performance of each candidate network structure. The fitness function can be based on the performance of the network on the validation set, such as accuracy, loss function value, etc. The fitness function can be expressed as:

$$f(c_i) = \text{Performance}(c_i) \tag{13}$$

where $f(c_i)$ is the fitness of the i-th chromosome and Performance() is the performance of the GCN network corresponding to the i-th chromosome on the validation set.

Based on the value of the fitness function, the chromosomes with high fitness are selected to go into the next generation. Among the selected chromosomes, two chromosomes are randomly selected for crossover operation to generate a new chromosome. This process can be represented as:

$$c_i' = \text{Crossover}(c_j, c_k) \tag{14}$$

where c_j and c_k are the two chromosomes selected for the crossover operation.

The newly generated chromosomes are subjected to mutational operations to increase the diversity of the population. This process can be represented as:

$$c_i'' = \text{Mutate}(c_i') \tag{15}$$

where c_i'' is the chromosome after the mutation operation and c_i' is the newly generated chromosome.

The newly generated chromosomes are replaced with the original chromosomes to form a new population. Specifically for the GCN model, the structure can be represented as:

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \tag{16}$$

It can be seen that we take the number of nodes, layers, and convolutional kernel size as the genes of chromosomes, evaluate the performance of each chromosome by the fitness function, and select the chromosome with high fitness to perform the crossover and mutation operations, and finally obtain the optimal GCN model structure.

4.2. **Commodity heterogeneous subgraph generation.** In this paper, we apply the principle of Graph Isomorphism Network (GIN) [31] to heterogeneous graphs and construct HetCodeNet. GIN is a method to do subgraph classification, and the principle is to classify subgraph representations by aggregating the node and structural information of the subgraphs to get subgraph representations.

Graph isomorphism detection using HetCodeNet does not require prior embedding of the knowledge graph; it is sufficient to extract the subgraphs, with one subgraph representing one commodity. The subgraphs are represented in a form similar to a neighbourhood table, as shown in Table 2. The first row is the number of nodes contained in the subgraph and the category to which the subgraph belongs. The following is the adjacency matrix, where the first number in each row indicates the node ID, the second number indicates the degree of the node, and the subsequent number indicates the number of points in the subgraph that the node is connected to.

Table 2. Subgraph representation

(Number of nodes)4 (Node ID)	(category) 6669 (Degree of node)	(Connected node number)
1109	0	1
14560	1	1
21267	0	2
24069	2	1

In order to generate the input for HetCodeNet, a commodity heterogeneous subgraph needs to be constructed based on the description text of the commodity. The first step is to extract the nodes by removing the blank characters at the beginning and end of the commodity description text as commodity nodes. The commodity description text is divided into multiple declaration elements according to the designed rules, the division rules are firstly divided by vertical lines, followed by semicolons and spaces, and the divided phrases are used as declaration element nodes. Finally, the keywords are extracted as keyword nodes by segmentation and other operations on the declared element nodes. The knowledge graph subgraph extraction process is shown in Algorithm 1.

For the declared element nodes that do not exist in the knowledge graph, they are segmented into keywords, and the keywords are used to go back to the knowledge graph to find the declared element node with the largest similarity. Eventually, this similar node is used instead of the declaration element node.

4.3. **Heterogeneous GA-GCN model construction.** After the subgraph is generated, it needs to be fed into the heterogeneous GA-GCN model for information prediction. The heterogeneous GA-GCN information prediction model is mainly divided into four parts, input layer, convolutional layer, pooling layer, and output layer, as shown in Figure 2. The input of the model is the heterogeneous subgraph of commodities, and the output is the predicted trade commodity information and the confidence of each information.

The commodity description is shown in Equation (17), where C^n denotes the nth commodity, D_i^n denotes the ith declared element of the nth commodity, and $W_{i,j}^n$ denotes the jth keyword of the ith declared element of the nth commodity.

$$C^n = \{D_1^n | D_2^n \dots\} = \{W_{11}^n W_{12}^n \dots | W_{21}^n W_{22}^n \dots\} \tag{17}$$

Confidence is a real number that describes the likelihood that a commodity belongs to a certain type. The likelihood of outputting information about multiple commodities for

Algorithm 1 Knowledge graph subgraph extraction

```
Input: text
Output: subgraph
Define nodes-dictionary;
Define commodity = text;
if commodity exists in nodes-dictionary then
    for declare in commodity all children do
        Add commodity-declare to subgraph;
        for keyword in declare all child nodes do
            Add declare-keyword to the subgraph;
        end for
    end for
else
    for declare in text do
        if declare exists in nodes-dictionary then
            for keyword in declare all children do
                Add declare-keyword to the subgraph;
            end for
        else
            Add the declare where the keyword is to declare-set;
            declare' = the most frequent declare in declare-set;
            for keyword in declare' all children do
                Add declare'-keyword to subgraph;
            end for
        end if
    end for
end if
Add the commodity where the declare is located to commodity-set;
commodity' = Commodity with the most occurrences in commodity-set;
for declare in commodity' all children do
    Add commodity'-declare to subgraph;
end for
return subgraph;
```

a single commodity description is called the confidence level, denoted by p, and is a real number between 0 and 1.

The problem of trade goods information prediction is defined as training a model f. For any good, its commodity description C^n is known, and the confidence that the good belongs to each label p^n can be obtained from the heterogeneous GA-GCN model, as shown in Equation (18).

$$f(C^n) = \{(Y_1, p_1^n), (Y_2, p_2^n), ...\} \tag{18}$$

where Y denotes the set consisting of all the labels.

The heterogeneous GA-GCN model input layer is used to receive commodity heterogeneous subgraphs. The subgraph is first read through the file and for the read subgraph, the node and edge information of the subgraph is stored into an object of type Graph by traversing each node and its neighbours. This converts the subgraph into the form of an adjacency matrix, which is later fed into the convolutional layer for subsequent operations.

Convolutional layer is the most important structure in HetCodeNet, the convolutional layer can learn the node features and structural features of subgraphs through a series of

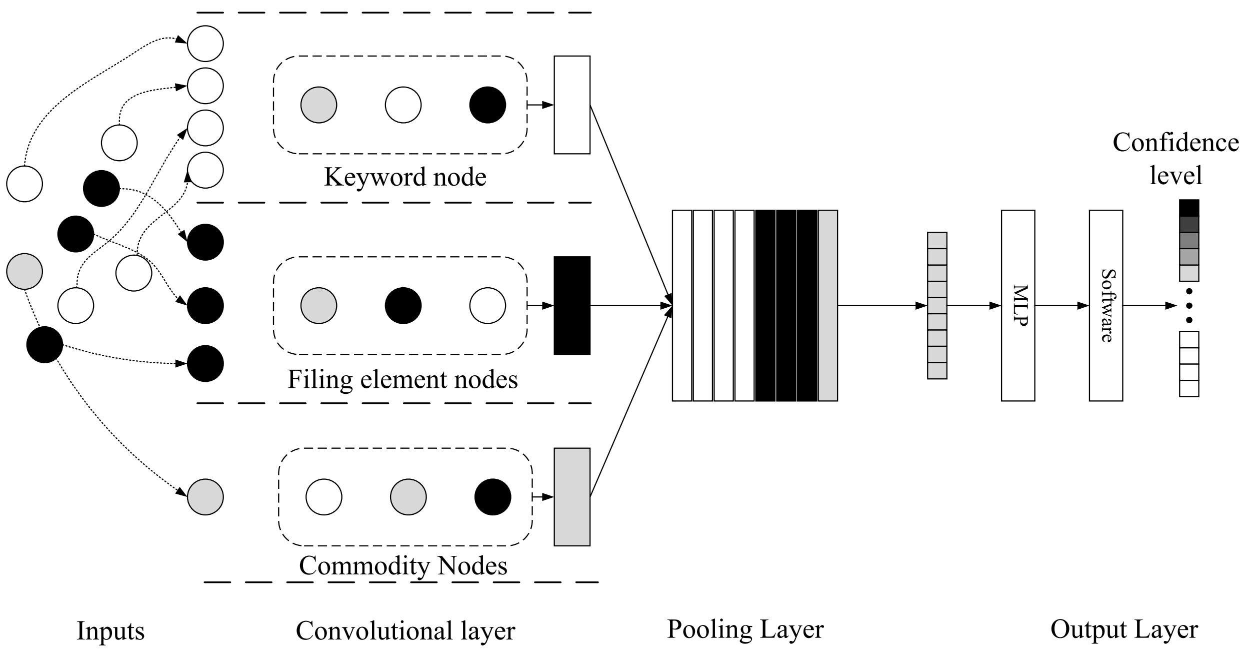

Figure 2. Heterogeneous GA-GCN information prediction models

operations such as aggregation, combination, and readout to the subgraph representation, and pass the subgraph representation to the lower layer for classification tasks. The convolutional layer of graph neural networks generally accomplishes two tasks, aggregation and combination. The aggregation process exists for many iterations, and each iteration can obtain an aggregated representation of the node's neighbourhood information. Combination is the combination of the aggregation result with the original representation of the node.

At the k-th iteration, the aggregation operation is shown in Equation (19).

$$a_v^{(k)} = \text{AGGREGATE}\left(\{h_u^{(k-1)} : u \in N(v)\}\right) \tag{19}$$

where $N(v)$ denotes the set of all neighbouring nodes of node v and $h_u^{(k-1)}$ denotes the feature vector of node u before the k-th iteration.

At the k-th iteration, the combination operation is shown in Equation (20).

$$h_v^{(k)} = \text{COMBINE}\left(h_v^{(k-1)}, a_v^{(K)}\right) \tag{20}$$

HetCodeNet uses a one-shot aggregation function, while the Multilayer Perceptron (MLP) can achieve one-shot aggregation, so the aggregation function in this paper uses the MLP.

$$h_v^k = \text{MLP}\left(\left(1+\epsilon^{(k)}\right) \cdot h_v^{(k-1)} + \sum_{u \in N(v)} h_u^{(k-1)}\right) \tag{21}$$

A READOUT function is also added to do the graph classification task, which serves to aggregate the node features to get a representation of the whole graph.

$$h_G = \text{READOUT}\left(\{h_v \mid v \in G\}\right) \tag{22}$$

READOUT can be a simple substitution-invariant function, such as summing, or a more complex graph-level pooling function.

The role of pooling layer is to further extract the subgraph features. The adjacency matrix of commodity subgraphs is processed by convolutional layer to get the feature vector representation of each commodity subgraph, and then the features are further extracted by the pooling layer, and the common pooling layers are MaxPool and AveragePool, etc. In this paper, we use AveragePool, the input is the commodity text feature matrix, and the average pooling extracts all the features in each subregion of the feature matrix. In this paper, we use AveragePool, the input of pooling layer is the commodity text feature matrix, and the average pooling extracts the average value of all the features in each sub-region of the feature matrix.

The role of the output layer is to map the text vector into the sample space. In order to make the output layer has the learning ability of nonlinear expression, the fully connected layer is often followed by an activation function, and the common activation functions are Sigmoid function, Tanh function, ReLU function. In this paper, the output layer consists of two fully connected layers, and the activation function is the ReLU function.

5. Experimental results and analyses.

5.1. **Experimental data.** In this work, a large coastal harbour is selected as a test object to validate the proposed heterogeneous GA-GCN model.

Import and export trade data of the port were collected for the period 2012 to 2021. The MATLAB programming language was used to design and implement the heterogeneous GA-GCN model and to automatically identify the import and export commodities of the port so as to predict the trade volume. Four categories of commodities were selected as the dataset, as shown in Table 3.

Table 3. Optimisation results of the four algorithms.

Data set	Sample size	Number of commodity information	Number of nodes	Number of ternary groups
Summer Men's Clothing Data	6968	194	23484	64271
Data on agricultural equipment	13354	537	76338	108686
Data on kitchen appliances	88390	328	44022	59242
Data on domestic vehicles	3603	139	13074	21140
Data on medical and health equipment	4517	169	24742	38257

In this work, data from 2012 to 2016 were used as training samples and the pre-processed data were tested using MATLAB. The data from 2012 to 2016 were used as input vector and the port cargo throughput from 2017 to 2021 were used as test samples so that the model created works best.

In this paper, information prediction is performed by heterogeneous GA-GCN model using node classification.GCN requires initial embedding representation of input nodes. The hyperparameters of the heterogeneous GA-GCN model are shown in Table 4.

5.2. **Evaluation metrics.** In this paper, Hits@8, Hits@4, Accuracy, Precision, Recall, and F1-score are used as the evaluation metrics for the task of predicting commodity customs codes.

Hits@n is the proportion of the number of samples with a confidence ranking of correct prediction less than or equal to n to the overall number of samples in the results of trade commodity forecasting. The specific calculation method is shown below:

$$\text{Hits@n} = \frac{1}{|C|}\sum_{i=1}^{|C|} 1(rank_i \leq n) \tag{23}$$

Where n denotes the number of results to be calculated. The n in this work is 8 and 4.

Accuracy is the ratio of the number of correctly predicted samples in the test set to the total number of samples in the test set. precision indicates how many of the positively predicted samples are truly positive samples. recall indicates how many of the truly

Table 4. Hyperparameters for Heterogeneous GA-GCN Models

Hyperparameterisation	Numerical value
Population Size	50
Encoding Length	10
Crossover Probability	0.7
Mutation Probability	0.01
Max Iterations	500
Node Embedding Dimension	6968
learning rate	13354
Number of Convolutional Layers	88390
MLP Layers	3603
Convolution method	4517

positive samples are correctly predicted to be positive samples. the F1-score takes into account both Precision and Recall.

5.3. **Commodity information prediction results.** In this paper, several knowledge graph embedding methods are used to compare with the proposed GA-GCN model, including TransE, PtransE, DistMult, R-GCN [32] and CompGCN [33].

TransE belongs to the traditional embedding methods, while PtransE adds path information on top of TransE.DistMult uses vector inner product to represent ternary relationships. R-GCN is based on KG2E which employs relationship-specific graph convolution operations.CompGCN employs a multitasking learning framework to jointly accomplish link prediction and entity classification.

The experimental results on summer men's clothing data are shown in Figure 3. It can be seen that the GA-GCN model achieved the best results for all evaluation metrics except Hits@8. CompGCN model is only 1.2

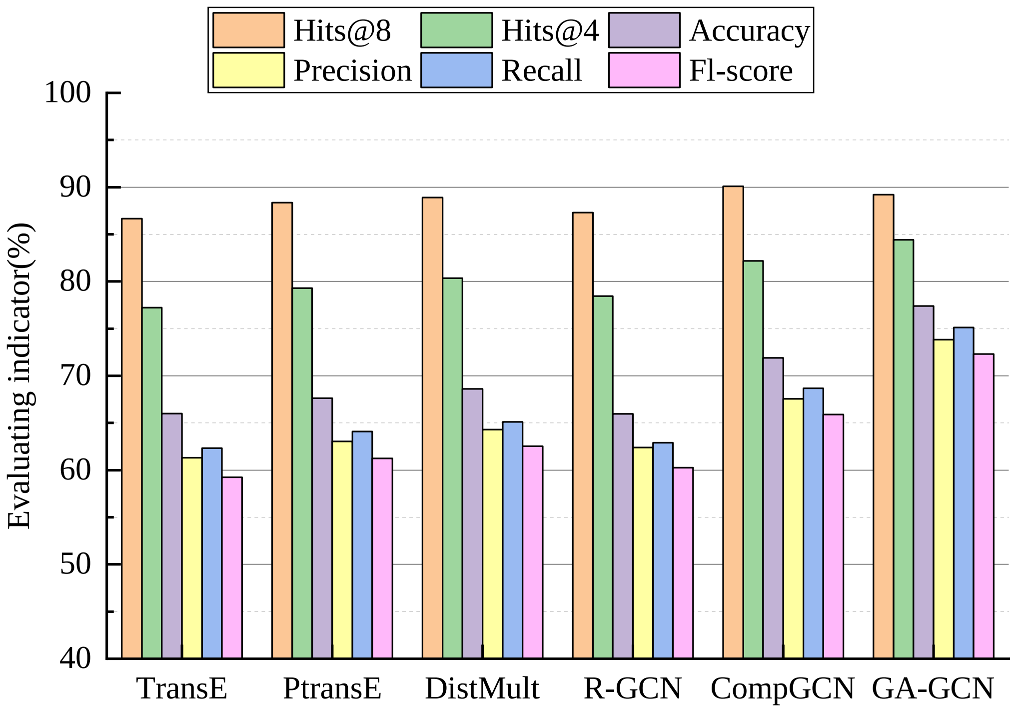

Figure 3. Experimental results on summer men's clothing data

The experimental results on the agricultural equipment data are shown in Fig. 4. It can be seen that the GA-GCN model achieved the best results for all evaluation metrics except Hits@4, which is the same as the results in Figure 3. CompGCN model is only 1.1% higher than GA-GCN model in Hits@4 evaluation index. The experimental results for

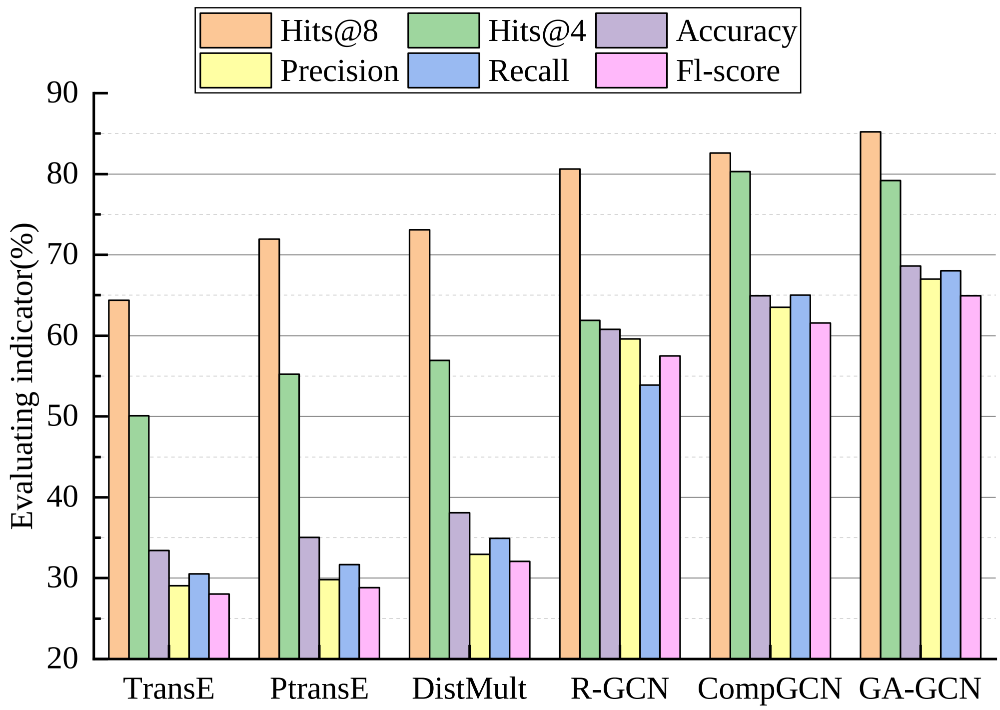

Figure 4. Experimental results on agricultural equipment data

the kitchen utensil data, household vehicle data and medical and health equipment data are basically consistent with Figure 4. Comprehensively analysing the prediction results on multiple datasets, GA-GCN achieves the best results on all other evaluation indexes of each dataset except on Hits@8 and Hits@4. This proves that the trade commodity knowledge graph constructed in this paper can abstract the knowledge in the commodity description text, and GA-GCN has a great advantage in learning the representation of heterogeneous subgraphs of commodities, which indicates that the subgraph classification idea of the GA-GCN model is very effective in the problem of trade commodity information recognition.

5.4. **Results of economic forecasting.** Finally, based on the identification of trade goods information, the trade volume prediction is continued. The already trained heterogeneous GA-GCN model is tested and the error analysis of the trained network model is performed. The predicted values after returning to normalisation are compared with the actual values, and the relative errors between the predicted values and the actual values are obtained, as shown in Table 5.

Table 5. Port Cargo Throughput Forecasts 2017 to 2021

	2017	**2018**	**2019**	**2020**	**2021**
Forecast (tonnes)	10371.58	10805.33	10722.09	10401.97	10681.26
Actual (tonnes)	10362.51	10814.51	10719.06	10404.86	10678.19
Relative error (%)	0.087	0.085	0.028	0.028	0.029

By comparing the predicted data with the actual data from 2017 to 2021, we find that the relative error between the predicted and actual values of cargo throughput is between 0.029% and 0.087%, with an average error of 0.006%. From the above analysis, we can see that the heterogeneous GA-GCN prediction model in this paper has high prediction accuracy and can be used to predict and analyse the cargo throughput of import and export trade in various large ports.

6. **Conclusion.** In this work, we designed a novel knowledge graph construction method and constructed a data knowledge graph of traded commodities based on containment relationships, so that multiple factors such as commodity codes, commodity attributes, transaction history, etc. can be considered comprehensively.GNNs can well cope with the problem of cross-modal information fusion. By connecting different types of nodes and edges into the same graph, GNNs can fuse different modalities together to achieve cross-modal information transfer and integration. In order to further improve the performance of the GNNs model, the GA algorithm is used to search for the optimal structure of the heterogeneous GNNs model, including the number of nodes, the number of layers, and the size of the convolution kernel. By defining the fitness function, the GA algorithm can search the optimal network structure among the candidate structures. Heterogeneous commodity information may involve multiple modalities, such as text, image, video, etc. However, the original GCN needs to compute the graph convolution of the whole graph at each layer, which has high computational complexity. Therefore, subsequent research will try to improve the computational efficiency by sample sampling and simplifying the computation.

REFERENCES

[1] G. Soava, A. Mehedintu, and M. Sterpu, "Analysis and Forecast of the Use of E-Commerce in Enterprises of the European Union States," *Sustainability*, vol. 14, no. 14, 8943, 2022.

[2] C. Tudor, "Integrated framework to assess the extent of the pandemic impact on the size and structure of the e-commerce retail sales sector and forecast retail trade e-commerce," *Electronics*, vol. 11, no. 19, 3194, 2022.

[3] T.-Y. Wu, J. C.-W. Lin, U. Yun, C.-H. Chen, G. Srivastava, and X. Lv, "An efficient algorithm for fuzzy frequent itemset mining," *Journal of Intelligent & Fuzzy Systems*, vol. 38, no. 5, pp. 5787-5797, 2020.

[4] T.-Y. Wu, L. Wang, and C.-M. Chen, "Enhancing the Security: A Lightweight Authentication and Key Agreement Protocol for Smart Medical Services in the IoHT," *Mathematics*, vol. 11, no. 17, 3701, 2023.

[5] T.-Y. Wu, Q. Meng, Y.-C. Chen, S. Kumari, and C.-M. Chen, "Toward a Secure Smart-Home IoT Access Control Scheme Based on Home Registration Approach," *Mathematics*, vol. 11, no. 9, 2123, 2023.

[6] C.-M. Chen, Y. Gong, and J. M.-T. Wu, "Impact of Technical Indicators and Leading Indicators on Stock Trends on the Internet of Things," *Wireless Communications and Mobile Computing*, vol. 2022, pp. 1-15, 2022.

[7] C.-M. Chen, S. Lv, J. Ning, and J. M.-T. Wu, "A Genetic Algorithm for the Waitable Time-Varying Multi-Depot Green Vehicle Routing Problem," *Symmetry*, vol. 15, no. 1, 124, 2023.

[8] N. Terzi, "The impact of e-commerce on international trade and employment," *Procedia-social and Behavioral Sciences*, vol. 24, pp. 745-753, 2011.

[9] A. A. Alyoubi, "E-commerce in developing countries and how to develop them during the introduction of modern systems," *Procedia Computer Science*, vol. 65, pp. 479-483, 2015.

[10] A. Alsaad, and A. Taamneh, "The effect of international pressures on the cross-national diffusion of business-to-business e-commerce," *Technology in Society*, vol. 59, 101158, 2019.

[11] Z. Xing, "The impacts of Information and Communications Technology (ICT) and E-commerce on bilateral trade flows," *International Economics and Economic Policy*, vol. 15, pp. 565-586, 2018.

[12] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61-80, 2008.

[13] D. Luo, W. Cheng, D. Xu, W. Yu, B. Zong, H. Chen, and X. Zhang, "Parameterized explainer for graph neural network," *Advances in Neural Information Processing Systems*, vol. 33, pp. 19620-19631, 2020.

[14] R. Chaplin-Kramer, M. Jonell, A. Guerry, E. F. Lambin, A. J. Morgan, D. Pennington, N. Smith, J. A. Franch, and S. Polasky, "Ecosystem service information to benefit sustainability standards for commodity supply chains," *Annals of the New York Academy of Sciences*, vol. 1355, no. 1, pp. 77-97, 2015.

[15] J.-S. Choi, W.-H. Lee, J.-H. Lee, J.-H. Lee, and S.-C. Kim, "Deep learning based NLOS identification with commodity WLAN devices," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 4, pp. 3295-3303, 2017.

[16] Q. Ji, E. Bouri, D. Roubaud, and L. Kristoufek, "Information interdependence among energy, cryptocurrency and major commodity markets," *Energy Economics*, vol. 81, pp. 1042-1055, 2019.

[17] Y. Luo, J. Ma, and C. Li, "Entity name recognition of cross-border e-commerce commodity titles based on TWs-LSTM," *Electronic Commerce Research*, vol. 20, pp. 405-426, 2020.

[18] K. Theofilatos, S. Likothanassis, and A. Karathanasopoulos, "Modeling and trading the EUR/USD exchange rate using machine learning techniques," *Engineering, Technology & Applied Science Research*, vol. 2, no. 5, pp. 269-272, 2012.

[19] B. Song, W. Yan, and T. Zhang, "Cross-border e-commerce commodity risk assessment using text mining and fuzzy rule-based reasoning," *Advanced Engineering Informatics*, vol. 40, pp. 69-80, 2019.

[20] J. Xu, Z. Hu, Z. Zou, J. Zou, X. Hu, L. Liu, and L. Zheng, "Design of smart unstaffed retail shop based on IoT and artificial intelligence," *IEEE Access*, vol. 8, pp. 147728-147737, 2020.

[21] H. Sun, X. Zhang, X. Han, X. Jin, and Z. Zhao, "Commodity image classification based on improved bag-of-visual-words model," *Complexity*, vol. 2021, pp. 1-10, 2021.

[22] X. Zou, L. Zhou, K. Li, A. Ouyang, and C. Chen, "Multi-task cascade deep convolutional neural networks for large-scale commodity recognition," *Neural Computing and Applications*, vol. 32, pp. 5633-5647, 2020.

[23] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimedia Tools and Applications*, vol. 80, pp. 8091-8126, 2021.

[24] A. Sohail, "Genetic algorithms in the fields of artificial intelligence and data sciences," *Annals of Data Science*, vol. 10, no. 4, pp. 1007-1018, 2023.

[25] J. Zhou, and Z. Hua, "A correlation guided genetic algorithm and its application to feature selection," *Applied Soft Computing*, vol. 123, 108964, 2022.

[26] A. A. Yusuf, F. Chong, and M. Xianling, "An analysis of graph convolutional networks and recent datasets for visual question answering," *Artificial Intelligence Review*, vol. 55, no. 8, pp. 6277-6300, 2022.

[27] H. T. Phan, N. T. Nguyen, and D. Hwang, "Aspect-level sentiment analysis: A survey of graph convolutional network methods," *Information Fusion*, vol. 91, pp. 149-172, 2023.

[28] C. Huang, M. Li, F. Cao, H. Fujita, Z. Li, and X. Wu, "Are graph convolutional networks with random weights feasible?," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 2751-2768, 2022.

[29] J. Liao, W. Zhou, F. Luo, J. Wen, M. Gao, X. Li, and J. Zeng, "SocialLGN: Light graph convolution network for social recommendation," *Information Sciences*, vol. 589, pp. 595-607, 2022.

[30] Z. Zhao, M. Tang, W. Tang, C. Wang, and X. Chen, "Graph convolutional network with multiple weight mechanisms for aspect-based sentiment analysis," *Neurocomputing*, vol. 500, pp. 124-134, 2022.

[31] X. Li, Y. Jiang, Y. Liu, J. Zhang, S. Yin, and H. Luo, "RAGCN: Region aggregation graph convolutional network for bone age assessment from X-ray images," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-12, 2022.

[32] M. Amagasaki, H. Oyama, Y. Fujishiro, M. Iida, H. Yasuda, and H. Ito, "R-GCN Based Function Inference for Gate-level Netlist," *IPSJ Transactions on System LSI Design Methodology*, vol. 13, pp. 69-71, 2020.

[33] W. Li, L. Ni, J. Wang, and C. Wang, "Collaborative representation learning for nodes and relations via heterogeneous graph neural network," *Knowledge-Based Systems*, vol. 255, 109673, 2022.