Mathematics and Computer Science
2022; 7(6): 124-129
http://www.sciencepublishinggroup.com/j/mcs
doi: 10.11648/j.mcs.20220706.14
ISSN: 2575-6036 (Print); ISSN: 2575-6028 (Online)

Application Research of Graph Neural Networks in the Financial Risk Control

Zhongbao Yu[*], Jiaqi Zhang, Xin Qi, Chao Chen

Data management and Application Department, Bank of Shanghai, Shanghai, China

Email address:
yuzhb@bosc.cn (Zhongbao Yu), zhangjq2@bosc.cn (Jiaqi Zhang), qixin1@bosc.cn (Xin Qi), chenchao6@bosc.cn (Chao Chen)
[*]Corresponding author

To cite this article:
Zhongbao Yu, Jiaqi Zhang, Xin Qi, Chao Chen. Application Research of Graph Neural Networks in the Financial Risk Control. *Mathematics and Computer Science*. Vol. 7, No. 6, 2022, pp. 124-129. doi: 10.11648/j.mcs.20220706.14

Received: April 8, 2022; **Accepted**: December 26, 2022; **Published**: December 28, 2022

Abstract: Combining deep learning with graph data, the method applied to learning tasks on association relationships is collectively referred to as Graph neural network (GNN). This paper mainly studies the application of GNN in the financial risk control. With the enterprise customer network graph, this paper designs a credit rating model based on GNN, an implicit relationship recognition model, and a fusion model of the two. To reduce duplication and improve model performance, graph pruning method is introduced in the data preprocessing stage, such as entity fusion, relationship normalization, etc. According to the prediction results, the heterogeneous graph credit rating model is better than the homogeneous one. Moreover, the suspicious relations detected by the implicit relation recognition model can be complementary to the heterogeneous graph credit rating model, which will improve the model performance. The model of this paper can be applied not only in the financial risk control, but also can provide a reference for other fields. In response to external public opinion information, the credit rating model label is effectively supplemented, and the heterogeneous graph credit rating model is used to learn related topology information, redefine the credit rating of related enterprises, leading to discover related high-risk enterprises, and achieve the purpose of risk control. This is an advantage that traditional machine learning methods do not have.

Keywords: Graph Neural Networks, Heterogeneous Graph, Graph Pruning, Risk Control

1. Introduction

In recent years, with the development of artificial intelligence, the transformation of Fintech has also faced challenges. Graph technology is a good area. Among them, knowledge graphs (KGs) are widely used in finance, biology, medicine, and many other application fields. Nodes in the graph represent entities, and edges represent relationships between entities. KGs are often accompanied by ontologies that specify the schemas that KGs follow, i.e., the types of entities and the relationships between different types [1]. Gong F et al. use embeddings to decompose drug recommendations into a link prediction process, taking into account both patient diagnoses and adverse drug reactions [2]. Patel A et al. analyze this COVID-19 knowledge graph and implement various algorithms to predict as yet undiscovered connections between concepts [3]. Bulla M et al. propose a proof-of-concept design for a financial knowledge graph, and a semantic question answering framework specifically for the financial domain [4]. This paper mainly studies the application of graphs in the financial field.

The financial risk control field generally collects and organizes a large amount of data with explicit or implicit correlations. Through these relationships, the originally isolated data records are connected to form a new type of graph-structured data. For example, the financial knowledge graph is also graph-structured data. However, the application of these graph-structured data that requires a lot of manpower and material resources is still in its infancy, and most of the current applications remain at the level of simple query and visualization of graph data. Machine learning algorithms for graph data, especially the GNN algorithm that has emerged in recent years, can deeply mine graph data [5]. The GNN continues to attract interest due to its good performance in various graph learning problems such as node classification and link prediction [6, 7]. It lays the foundation for innovation in banking and finance. Since Google first proposed the "knowledge graph", it has been widely used in various fields. Due to the correlated nature of graph data, it has unique advantages in identifying gang fraud and risk transfer. Kurshan E et al. discuss the implementation difficulties faced by current and next-generation graph solutions [8]. The credit approval process applied to the financial industry shifts from the traditional expert experience risk control model to big data risk control and artificial intelligence risk control.

Knowledge graphs combined with graph computing applications, traditional graph algorithms, and reliability can be used to mine customer association paths, important customers, and community groups. Zhang L introduces the theory and structural parsing of knowledge graphs [9]. The application of graph algorithms is mainly studied in biology [10, 11]. The application of reliability in the graph is mainly discussed in [12, 13].

The traditional supervised machine learning risk control model only performs mathematical modeling from the customer dimension, and does not consider the relationship between customers to build the model. GNNs can build supervised learning models that consider associations. Graph-based semi-supervised learning, Ye W et al. study the over-smoothing problem [14].

As a model that can reveal deep topological information, GNN mainly studies the application of the graph network, and proposes further research questions in the future.

This paper introduces the application of GNN in the financial risk control field, and also provides a reference for other fields. The rest of this paper is arranged as follows. Section 2 introduces the design of schema. Section 3 analyzes the model. Section 4 shows the application of GNN. The last section concludes the paper.

2. The Design of Schema

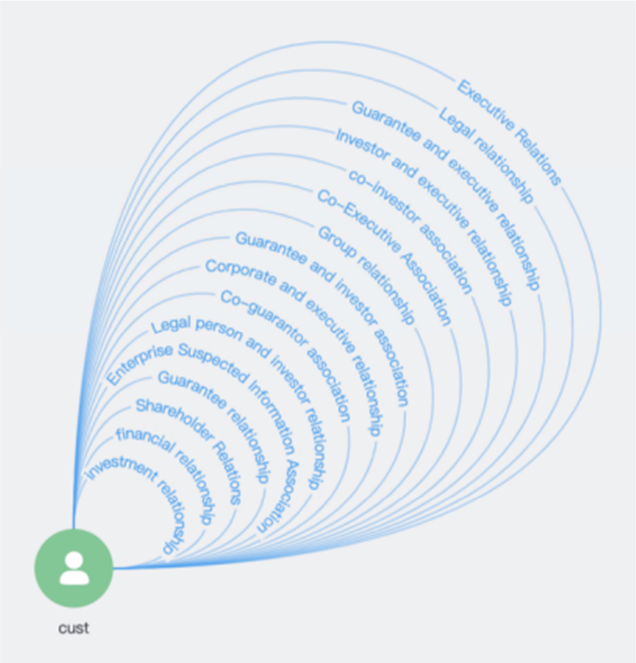
Executive Relations
Legal relationship
Guarantee and executive relationship
Investor and executive relationship
co-investor association
Co-Executive Association
Group relationship
Guarantee and investor association
Corporate and executive relationship
Co-guarantor association
Legal person and investor relationship
Enterprise Suspected Information Association
Guarantee relationship
Shareholder Relations
financial relationship
investment relationship
cust

Figure 1. *Schema.*

The core of the knowledge graph is the design of the schema. In the ontology model, concepts, attributes, and relationships between concepts need to be constructed. The knowledge modeling process is the foundation of knowledge graph construction. A high-quality data model can avoid a lot of unnecessary and repetitive knowledge acquisition work, effectively improve the efficiency of knowledge graph construction, and reduce the cost of domain data fusion. Knowledge in different fields has different data characteristics, and different ontology models can be constructed respectively. For a financial institution's public relationship data, manual modeling is used. This paper constructs one entity of customer and more than ten kinds of relationships including shareholder relationships, investment relationships, capital relationships and guarantee relationships. The schema is presented in the Figure 1. In the following practical applications, unless otherwise specified, this ontology model is used for design.

3. The Analysis of Model

GNNs can be regarded as an extension of deep learning in non-Euclidean space, which has developed rapidly in recent years. As a representative of the derivation graph learning method, GraphSAGE contains two steps during training: neighbor sampling and feature aggregation [15]. GraphSAGE is widely used in various fields [16-18]. During training, the neighbor nodes of each node are randomly selected to construct the corresponding subgraph, and the weight matrix of the same layer of all subgraphs is shared. Different aggregation methods can be selected when aggregating neighbors, such as minimum, maximum, average, etc. When training, instead of adding new nodes to the graph to update the entire graph and retrain, GraphSAGE builds subgraphs based on the new nodes, and then directly computes predictions using the parameters of each layer obtained from training. GraphSAGE is mostly used in homogeneous graph scenarios, and the RGCN can be used for heterogeneous graph modeling [19]. And the GCN is widely used in various fields [20-23]. This section mainly discusses the forward pass of the GraphSAGE and the RGCN.

3.1. The Introduction of the GraphSAGE

The character description of the GraphSAGE is shown in Table 1, and the pseudocode is shown in Table 2.

Table 1. *The character description of the GraphSAGE.*

step	content
input	G(V, E) where G represents the graph, V represents the set of graph nodes and E represents the set of graph edges; Deepth: K; Weight matrix W^k, where $\forall k \in \{1, \ldots, K\}$; activation function σ; Aggregate function AGGREGATE_k, $\forall k \in \{1, \ldots, K\}$; Neighbor sampling rate S_k
output	Embedding z_v, where $\forall v \in V$

Table 2. *The pseudocode.*

```
$h_v^0 \leftarrow x_v, \forall v \in V$
for $k = 1...K$ do
    for $v \in V$ do
        $h_{N(v)}^k \leftarrow \text{AGGREGATE}_k(\{h_u^{k-1}, \forall u \in N(v)\})$
        $h_v^k \leftarrow \sigma(W^k \text{CONCAT}(h_u^{k-1}, h_{N(v)}^k))$
    End
    $h_v^k \leftarrow h_v^k / ||h_v^k||_2, \forall v \in V$
end
$z_v \leftarrow h_v^K, \forall v \in V$
```

Example 1. Take Figure 2 as an example to explain GraphSAGE embedding and set K = 1, σ = RELU,

$$\text{AGGREGATE} = \text{mean }(),\ W^0 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix},\ S_1 = 3.$$

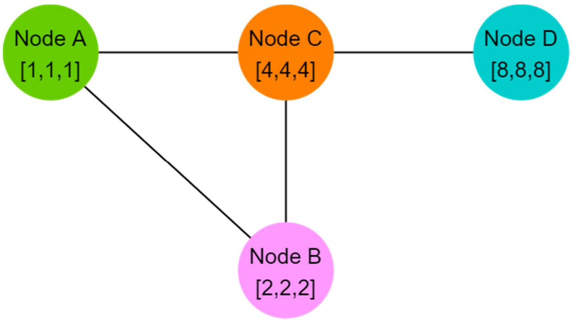

Figure 2. *GraphSAGE Embedding.*

Figure 2 is an undirected graph of 4 nodes A, B, C, D, and the attributes of the 4 nodes are [1,1,1], [2,2,2], [4,4,4] and [8,8,8], respectively.

$h_A^0 = [1,1,1],\ \ h_B^0 = [2,2,2],\ \ h_C^0 = [4,4,4],\ \ h_D^0 = [8,8,8]$

$h_{N(A)}^1 = \text{mean } (h_B^0 + h_C^0) = [3,3,3]$

$h_{N(B)}^1 = \text{mean } (h_A^0 + h_C^0) = [2.5,\ \ 2.5,\ \ 2.5]$

$h_{N(C)}^1 = \text{mean } (h_A^0 + h_B^0 + h_D^0) = [3.7,\ \ 3.7,\ \ 3.7]$

$h_{N(D)}^1 = \text{mean}(h_C^0) = [4,4,4]$

$h_A^1 = \sigma((W^0)^T \text{ CONCAT}(h_A^0,\ h_{N(A)}^1)^T) = [12,\ 12,\ 12,\ 12]^T$

$h_B^1 = \sigma((W^0)^T \text{ CONCAT}(h_B^0,\ h_{N(B)}^1)^T) = [13.5,\ 13.5,\ 13.5,\ 13.5]^T$

$h_C^1 = \sigma((W^0)^T \text{CONCAT}(h_C^0,\ h_{N(C)}^1)^T) = [22,\ 22,\ 22,\ 22]^T$

$h_D^1 = \sigma((W^0)^T \text{CONCAT}(h_D^0,\ h_{N(D)}^1)^T) = [36,\ 36,\ 36,\ 36]^T$

At this time, the node representations of h_A^1, h_B^1, h_C^1 and h_D^1 are obtained, and the node labels are added, which can be used as supervised learning samples for training.

3.2. The Introduction of the RGCN

The GraphSAGE is mainly used in isomorphic graph scenarios. An isomorphic graph means that there is only one node type and one relationship type in the graph. In practical financial application scenarios, there are many types of nodes and relationships, which belong to the category of heterogeneous graphs. For heterogeneous graph modeling, the RGCN can be used.

RGCN also uses neighbor nodes as the receptive field of the target node, and generates the expression of the target node by aggregating neighbor features. The RGCN update formula is Eq. (1), and the formula description is shown in Table 3.

$$h_i^{(l+1)} = \sigma(\sum_{r \in R} \sum_{j \in N_i^r} \frac{1}{c_{i,r}} W_r^{(l)} h_j^{(l)} + W_0^{(l)} h_i^{(l)}) \qquad (1)$$

Table 3. *The character description of RGCN.*

$h_i^{(l)}$ represents the representation of node v_i at level l;
R represents the relationships in a heterogeneous graph where N_i^r represents the set of neighbor nodes associated with node i through relation r;
$c_{i,r}$ represents the normalization constant.

Example 2. Take Figure 3 as an example to explain RGCN and set σ = RELU, $c_{i,r} = |N_i|$,

$$W_0^0 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},\ W^0_{\text{colleague}} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},\ W^0_{\text{schoolmate}} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},\ W^0_{\text{couple}} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

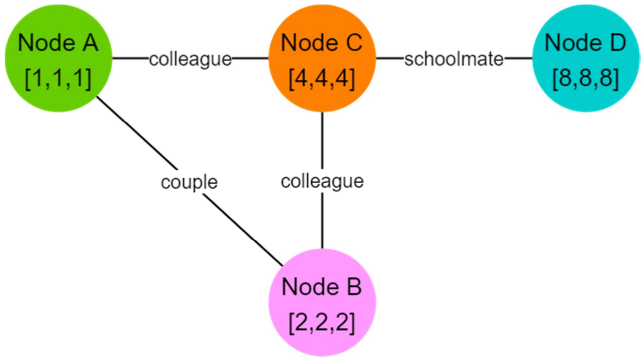

Figure 3. *GCN Embedding.*

$h_A^0 = [1,1,1],\ \ h_B^0 = [2,2,2],\ \ h_C^0 = [4,4,4],\ \ h_D^0 = [8,8,8]$

$$h_C^1 = \sigma\ (0.5(h_A^0\, W^0_{\text{colleague}} + h_B^0\, W^0_{\text{colleague}}) + h_D^0\, W^0_{\text{schoolmate}} + h_C^0 W_0^0)$$

$$= \sigma(0.5(\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} [1,1,1]^T + \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} [2,2,2]^T) + \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} [8,8,8]^T + \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} [4,4,4]^T)$$

$$= [40.5,\ \ 40.5,\ \ 40.5,\ \ 40.5]^T$$

The result shows that there is a difference from $h_C^1 = [22, 22, 22, 22]^T$ in Example 1.

4. The Application of GNN

One common application of GNN is anti-fraud in credit cards. This business scenario can be abstracted as a binary classification problem in machine learning: good customers can borrow money, while bad customers cannot. Normally the financial institutions applied some normal machine learning models like Logistic regression to it, but the performance of these models is not satisfactory. This paper mainly studies this problem from the perspective of graph models, using graph learning to identify the quality of customers and decide whether to lend money to applying customers.

Table 4 shows the modeling steps of GNN. This paper uses the schema of Figure 1 and the remaining three steps are introduced in the following content.

Table 4. *The model steps.*

Step1. Build a schema that conforms to business logic;
Step2. Graph pruning operation based on schema;
Step3. Extract subgraphs for invisible relationship prediction;
Step4. Credit Rating Prediction Based on Supplementary Predictive Relationships.

4.1. Graph Pruning

According to the constructed association graph, this paper removes the nodes whose names have no business meaning with the highest degrees, such as "property, education bureau, labor union, stock", etc.

By entity integration, company names similar to "company group and company (group)" will be normalized to "company group", reducing business ambiguity and closer to the actual business.

4.2. The Classification and Link Predict of Graph SAGE

The graph data is achieved from the guarantee relationship subgraph of the financial institution PR graph. It is a homogeneous graph with one sponsorship relationship, over 30,000 nodes and nearly 50,000 edges. Among them, the number of nodes with positive labels is about 5000, and the number of nodes with negative labels is about 200. Using the GraphSAGE training, the model result AUC can reach 0.7127, which belongs to the available category. In order to enrich the relationship data and improve the model performance, this section will introduce the GraphSAGE Link Predict model.

The GraphSAGE Link Predict model studies the "probability of not directly associating customers with links." This model adopts a test set of 100 positive samples and 300 negative samples, and then performs link prediction based on more than 18,000 nodes and 15,000 pairs of relationships.

In Figure 4, there is no direct relationship between customer 1 and customer 3. The model predicts that the probability of a relationship is 1. It is verified offline that customer 1 and customer 3 are truly related.

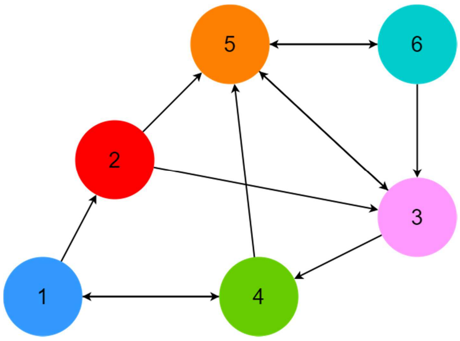

Figure 4. *Link prediction graph.*

Based on the results predicted by GraphSAGE LP, this paper establishes the relationship between customer 1 and customer 3, as shown in Figure 5.

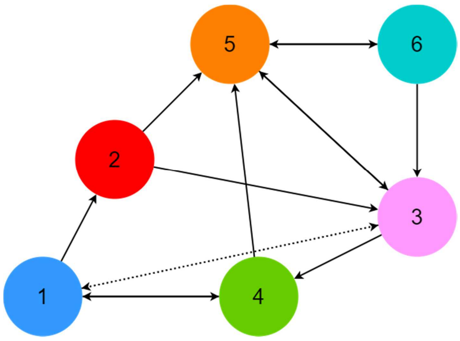

Figure 5. *Link Prediction Completion Graph.*

After adding the results predicted by the GraphSAGE LP model to the GraphSAGE Classification model data, the AUC of the model results increases to 0.7349 (three percent increase compared to the initial).

Furthermore, considering that the edges in real business scenarios always contain multiple types and belong to heterogeneous graphs, the RGCN classification model can be used, and the predicted invisible relations are supplemented in the original graph.

4.3. The Classification of RGCN

The data is taken from the public relation network with over 4,000,000 edges. For LightGBM and RGCN models, five features are used, including debenture_over_cnt_sum, putout_dt_mons, debenture_over_cnt_last_1y, lvl_worst and scale_type.

At the same time, customers are divided into two categories: internal and external, which are in line with actual business; effectively improve the effect of the model. RGCN's AUC is six percent higher than LightGBM's.

Table 5. *Model field.*

Model input fields	Field meaning
putout_dt_mons	The number of months between the latest IOU loan time and the observation time point
debenture_over_cnt_sum	IOUs with overdue history
debenture_over_cnt_last_1y	Number of overdue IOUs issued in the previous year
lvl_worst	Worst loan rating ever
scale_type	Enterprise size

4.4. Model Analysis

For customers, there are the following four business situations:

1. The customers have no attribute information, no relationship, no uniqueness identification, and no commercial value;
2. The internal customers have attribute information, but their relationships cannot be obtained. For such a client, a self-connecting edge can be added, and RGCN can also handle it;
3. For external customers, there is no attribute information, but they have relationships instead. Using 100 epochs, the AUC on the test set can reach 0.7350, and the ks can reach 0.4032;
4. Internal customers have attribute information and related relationships. RGCN consistently outperforms LightGBM models on the test set.

Table 6. *Model Comparison.*

Model	1	2	3	4
LightGBM	X	√	X	√
RGCN	X	√*	√	√*

*: Add a self-loop to each node to deal with it

*: RGCN performs better

5. Conclusion

This paper mainly studies the application of the GNN in financial institution risk control, credit rating identification, indirect customer link prediction and so on. GraphSAGE and RGCN algorithms are respectively used to establish the credit rating recognition model of the homogeneous graph. The result of the link prediction model improves the credit level recognition, and the final AUC reaches 0.7349; the credit level recognition model of the heterogeneous graph is 6% better than LightGBM. Application results show that RGCN outperforms GraphSAGE and existing LightGBM models. The improved RGCN model can be combined with the current LightGBM model.

References

[1] Kaoudi Z, Lorenzo A, Markl V. Towards Loosely-Coupling Knowledge Graph Embeddings and Ontology-based Reasoning [J]. 2022.

[2] Gong F, Wang M, Wang H, et al. SMR: Medical Knowledge Graph Embedding for Safe Medicine Recommendation [J]. Big Data Research, 2021, 23: 100174.

[3] Patel A, Pai S S, Rajamohan H R, et al. Finding Novel Links in COVID-19 Knowledge Graph Using Graph Embedding Techniques [C]// Smoky Mountains Computational Sciences and Engineering Conference. Springer, Cham, 2022.

[4] Bulla M, Hillebrand L, Lübbering, Max, et al. Knowledge Graph Based Question Answering System for Financial Securities [J]. 2021.

[5] Liu Y, Zhang M, Ma C, et al. Graph neural network [J]. 2020.

[6] Chen Z, Ma T, Y Wang. When Does A Spectral Graph Neural Network Fail in Node Classification? [J]. 2022.

[7] Hanik M, Demirta M A, Gharsallaoui M A, et al. Predicting Cognitive Scores With Graph Neural Networks Through Sample Selection Learning [J]. 2021.

[8] Kurshan E, Shen H. Graph Computing for Financial Crime and Fraud Detection: Trends, Challenges and Outlook [J]. International Journal of Semantic Computing, 2020, 14 (04): 565-589.

[9] Zhang L. Knowledge graph theory and structural parsing [J]. university of twente, 2002.

[10] Gainullina A N, Shalyto A A, Sergushichev A A. Method of the Joint Clustering in Network and Correlation Spaces [J]. Modeling and Analysis of Information Systems, 2020, 27 (2): 180-193.

[11] Gainullina A N, Shalyto A A, Sergushichev A A. Method for Joint Clustering in Graph and Correlation Spaces [J]. Automatic Control and Computer Sciences, 2022, 55 (7): 647-657.

[12] Zhongbao Y, Fangming S, Zuyuan Z. Researches for more reliable arrangement graphs in multiprocessor computer system [J]. Applied Mathematics and Computation, 2019, 363: 124611.

[13] Zhongbao Y, Fangming S. Approximation Algorithm of Arrangement Graph Reliability in Parallel System [J]. Journal of East China University of Science and Technology, 2020, 46 (6): 838-842.

[14] Ye W, Huang Z, Hong Y, et al. Graph Neural Diffusion Networks for Semi-supervised Learning [J]. 2022.

[15] Hamilton W L, Ying R, Leskovec J. Inductive Representation Learning on Large Graphs [J]. 2017.

[16] Chang L, Branco P. Graph-based Solutions with Residuals for Intrusion Detection: the Modified E-GraphSAGE and E-ResGAT Algorithms [J]. 2021.

[17] Chen Z, Deng Q, Zhao Z, et al. Energy consumption prediction of cold source system based on GraphSAGE. 2021.

[18] Hajibabaee P, Malekzadeh M, Heidari M, et al. An Empirical Study of the GraphSAGE and Word2vec Algorithms for Graph Multiclass Classification [C]// 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). IEEE, 2021.

[19] Schlichtkrull M, Kipf T N, Bloem P, et al. Modeling Relational Data with Graph Convolutional Networks [C]. European Semantic Web Conference. Springer, Cham, 2018.

[20] Sun D, Ma L, Ding Z, et al. An Attention-Driven Multi-label Image Classification with Semantic Embedding and Graph Convolutional Networks [J]. 2022.

[21] Y Yao, Joe-Wong C. FedGCN: Convergence and Communication Tradeoffs in Federated Training of Graph Convolutional Networks [J]. 2022.

[22] Niu H, Haitao H E, Feng J, et al. Knowledge Graph Completion Based on GCN of Multi-Information Fusion and High-Dimensional Structure Analysis Weight [J]. Chinese Journal of Electronics, 2022.

[23] J Gao, X Liu, Chen Y, et al. MHGCN: Multiview Highway Graph Convolutional Network for Cross-Lingual Entity Alignment [J]. Tsinghua Science and Technology, 2022, 27 (4): 719-728.