A Cascade GCN-LightGBM Framework for Abnormal Account Detection in Social Networks

Bin Ni
Department of Information Technology, Henan Judicial Police Vocational College, Zhengzhou 450000, China
E-mail: BinnNii@outlook.com

Keywords: Graph convolutional network, LightGBM, social networks, topological relation

Received: August 5, 2025

We propose a two-stage cascade pipeline that first extracts structural embeddings via a two-layer Graph Convolutional Network (GCN) and then fuses them with raw node attributes for LightGBM classification to detect abnormal accounts in social networks. Evaluated on three datasets—Weibo-social (1.26 M nodes, 18.4 M edges), NSL-KDD and IoT-23—the framework attains an F1 of 88.5 % on Weibo-social, outperforming single GCN (87.9 %) by 0.6 pp and XGBoost (85.5 %) by 3.0 pp while reducing training time from 247.8 min to 12.3 min (≈20× acceleration). Cross-domain transfers show consistent gains of 1.8–2.1 pp F1, confirming the synergy of topological and attribute features. Compared with existing behaviour-, content- or single-graph baselines, our approach simultaneously improves accuracy and efficiency without extra annotations.

Povzetek: Študija predstavi dvostopenjski postopek, ki z dvoslojno GCN izdela strukturne vgraditve in jih združi z atributi vozlišč za klasifikacijo z LightGBM, s čimer učinkovito zazna abnormalne račune z izkoriščanjem sinergije topoloških in vsebinskih značilk.

1 Introduction

As the digital network ecology continues to evolve, network account security [1] has become a key link in guarding users' rights and interests and network order. Among them, the hazards brought by abnormal accounts are becoming more and more prominent [2]: zombie accounts, with the help of automated scripts and batch operations, massively disseminate false information on social platforms, information communities and other scenarios, ranging from misleading public opinion and creating panic to maliciously attracting traffic and committing fraud, eroding the authenticity of network information in all aspects [3]; Sybil accounts (witch attack accounts) [4], however, infiltrate into normal user groups or launch phishing attacks to steal privacy by forging identities and constructing false social networks, or launching phishing attacks to steal privacy. Sybil accounts (witch attack accounts) [4], on the other hand, by forging identities, constructing false social networks, infiltrating into normal user groups, or launching phishing attacks to steal privacy [5], or engaging in illegal behaviors such as cyber water armies [6] and malicious bill-sweeping, are a serious threat to the security of users' information and the health of the platform's ecosystem.

However, the detection of anomalous accounts faces multiple challenges. On the one hand, the dynamic nature of features increases the difficulty of defence - attackers are well aware of the traditional detection system's reliance on static features (e.g., fixed login IPs, commonly used device identifiers). By dynamically changing proxy IPs and simulating multi-device environments, they can quickly bypass the detection rules and invalidate the defence system constructed based on static features [7, 8].

On the other hand, network complexity further blurs the boundary of anomalies - on crowdsourcing platforms, real people and machine accounts are mixed for task execution, short URL services hide the real jump links, and the diversified network behaviors of normal users (e.g., cross-platform sharing, temporary short-link usage) highly overlap with the malicious operations of anomalous accounts in terms of their surface features, making it difficult to differentiate them accurately. It is difficult to differentiate them accurately; in addition, the computational efficiency problem has become a bottleneck for large-scale detection - the traditional model requires full-volume data processing. In the face of hundreds of millions of accounts and terabytes of behavioural logs, it not only consumes a long time, but also results in a great waste of storage and arithmetic resources, making it difficult to adapt to the actual needs of real-time monitoring and rapid response [9]. These challenges are intertwined with each other, pushing academia and industry to explore more efficient and intelligent anomalous account detection solutions continuously.

In the field of anomalous account detection, existing methods have significant limitations. Behavioural or content-based models are highly dependent on manually designed features, such as manually refining the login frequency of an account, keywords of posted content, etc. [10-12]. However, the network environment is complex and volatile, new abnormal behavioural patterns keep emerging, and artificial features are difficult to cover these dynamic changes in time, resulting in a weak generalisation ability of the model when facing unseen scenarios, and unable to identify new abnormal accounts effectively. The single graph model in detection [13]

focuses on mining topological relationships between accounts, but ignores the account's attribute features, such as account registration time, historical reputation rating, etc. [14]. At the same time, unsupervised detection methods lack label guidance, it is difficult to distinguish between normal and abnormal account boundaries accurately, and in the face of complex confusion scenarios, the detection accuracy is difficult to meet the actual needs, prone to omission and misjudgment, and unable to guard network information security [15-17] reliably.

To address the pain points of anomalous account detection [18], this paper makes a series of key contributions: proposes the GCN-LightGBM cascade framework, which skillfully integrates topology and attribute features, breaks the limitations of a single model, and comprehensively captures the anomalous patterns of the account; designs a lightweight distributed training process, and reasonably splits the scheduling of the data and the tasks, to adapt to the massive data scenarios, and solve the resource wasting problem of the traditional model's full processing; also builds a test environment that simulates real dynamic changes, and experimentally verifies that the framework shows good robustness in terms of training time and resource consumption. A controlled testbed is constructed to emulate evolving attack patterns, validating the framework's robustness against concept drift, and highlights the efficiency advantage in terms of training time and resource consumption, thus building a solid performance foundation for actual deployment.

We treat abnormal-account detection as a binary classification task on an attributed social graph $G_t = (V_t, E_t, X_t)$ that evolves daily. The goal is to learn $f : V_t \to \{0,1\}$ maximising F1 while keeping training time below 15 min on a 32-core server. We hypothesise: H1 — adding GCN structural embeddings significantly lowers the 10-fold CV error of LightGBM (paired t-test, $p<0.01$); H2 — under equal GPU-hour budget, the two-stage cascade reaches minimum validation loss in fewer epochs than an end-to-end GNN of comparable width; H3 — the gains in H1–H2 remain statistically significant across a 30-day concept-drift window. Experiments in Section 4 quantify each hypothesis and identify boundary conditions.

Although the primary focus is abnormal-account detection in social networks, we additionally evaluate the proposed framework on two non-social graph datasets — NSL-KDD and IoT-23 — to assess its cross-domain structural generalisation capability. No social-media-specific features (e.g., tweets, avatars) are used in these auxiliary experiments.

2 Related theoretical knowledge

2.1 Graph convolutional networks (GCN)

In the technical system of anomalous account detection, Graph Convolutional Network (GCN) is one of the key theoretical supports. Its core principle revolves around the neighbour aggregation mechanism to achieve the updating of node representations, which is mathematically portrayed by equation (1) [19].

$$H^{(k+1)} = PReLU(\hat{A}\hat{D}^{-1}H^{(k)}W^{(k)} + XW_{dp}) \quad (1)$$

Here, $\hat{A}$ is the normalized adjacency matrix, which assumes the role of combing the node connection relationship in the network and eliminating the influence of the difference in connection strength, so that the node interactions in different connection density regions can be calculated under a unified scale; X represents the initial features, covering the account registration information, the basic behavioral trajectory and other raw data; and W_{dp} is the residual connection weight, which is the residual structure that can effectively alleviate the problem of the disappearance of the gradient during the training of the deep network With the help of the residual structure, it can effectively alleviate the problem of gradient disappearance during deep network training, guaranteeing the integrity of information transfer and the stability of model training. In terms of practical application value, GCN has the unique advantage of being able to capture abnormal subgraph structures accurately. In network account scenarios, such as densely connected Spam (spam) account clusters [20], these abnormal accounts often form tight subgraphs by interconnecting with each other. GCN can identify such abnormal aggregation patterns based on the aggregation and analysis of the graph structure from the massive account relationships, which can provide key structural features for the subsequent detection of abnormal accounts [21], and help build up a strong technical defence line for network information security. The interlayer propagation formula of GCN is equation (2).

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

where $H^{(l)}$ is the node feature matrix of layer l and $\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}$ is a symmetric normalization of the adjacency matrix [22] to avoid feature scale imbalance due to node degree differences. The computation of $\hat{A}$ is shown in Equation (3).

$$\hat{A} = A + I_N \quad (3)$$

where A is the original adjacency matrix and I_N is the unit matrix. The GCN encoder consists of two graph convolutional layers with 128-dimensional hidden units, PReLU activation, and a dropout rate of 0.3. Weights are initialised using Xavier uniform distribution.

2.2 Core algorithm and optimization mechanism of lightgbm model

LightGBM is an efficient integrated learning framework based on GBDT, developed by Microsoft as an open-

source project. It uses histogram optimization technology to discretize continuous features to build a histogram, significantly reducing the computation and memory consumption of the decision tree to find the optimal split point; through the gradient one-sided sampling, according to the sample gradient size ranking, retaining the high-gradient samples and randomly sampling the low-gradient samples, to improve the training efficiency without affecting the accuracy. In addition, optimization techniques such as leaf growth strategy and mutually exclusive feature bundling make it a powerful tool for data processing in large-scale data scenarios such as recommender systems and financial risk control [23].

In the field of integrated learning, LightGBM serves as an efficient implementation of gradient boosting decision trees (GBDT) based on gradient boosting. Its prediction model is constructed following the core logic of gradient boosting [24]. Specifically, the prediction model formalisation of LightGBM is defined in equation (4).

$$F_T(x) = f_0(x) + \sum_{t=1}^{T} \alpha_t f_t(x) \quad (4)$$

where $f_0(x)$ represents the initial weak learner, e.g., a single-layer decision tree, $f_t(x)$ represents the prediction function for the t-th tree, and α_t is the learning rate, which controls the contribution weight of each tree..

Calculation of splitting gain and histogram optimisation in the LightGBM model, the splitting gain of the decision tree is shown in equation (5).

$$Gain = \frac{1}{2}\left[\frac{(\sum_{i\in L} g_i)^2}{\sum_{i\in L} h_i + \lambda} + \frac{(\sum_{i\in R} g_i)^2}{\sum_{i\in R} h_i + \lambda} - \frac{(\sum_{i\in P} g_i)^2}{\sum_{i\in P} h_i + \lambda}\right] - \gamma \quad (5)$$

where g_i , h_i are the first-order and second-order gradients of the loss function of sample i. λ is the L2 regularization coefficient to prevent overfitting. γ is the split threshold to control the tree complexity. LightGBM designs two core strategies for efficient training [25]: gradient one-sided sampling is based on the difference of sample gradients, retaining key samples with large absolute gradient values and randomly discarding samples with small gradients, which accelerates training and maintains the accuracy at the same time; and mutual exclusion feature binding is used for mutually exclusion features with different non-zero values at the same time and combines them into a single feature to reduce the feature dimensions and reduce the computational overhead, which is suitable for high-dimensional and sparse data scenarios. This reduces the feature dimension and computational overhead, and is suitable for high-dimensional sparse data scenarios.

2.3 Fusion model

Under complex data scenarios and diversified task requirements, its own capacity boundaries often limit a single model, and the construction of fusion models has become a key path to break through performance bottlenecks. GCN focuses on mining the structural relationships of the data, modelling the dependency between nodes with the help of graph structure, which can effectively capture the topological level of correlation information. However, there is a shortcoming in the utilisation of content features, and it lacks detailed and comprehensive processing capabilities for the attributes, semantic features, and other content features that are carried by the nodes or samples themselves [26]. However, it has shortcomings in the utilisation of content features. It lacks of detailed and comprehensive processing ability for attributes, semantics and other content features carried by nodes or samples themselves [26]; LightGBM, as a highly efficient gradient enhancement framework, is good at fast processing and accurate fitting of attribute data, and efficiently exploits feature values by histogram optimisation and other techniques, but completely ignores the topological structure of the data. It is difficult to capture the correlations of the samples or nodes in the graph structure.

The complementary characteristics of the advantages of different models allow the fusion model to show a strong potential. In the field of chemistry, in the practice of fusing GCN and a large language model (LLM) [27], GCN mines the topological associations of molecular structure. LLM understands the chemical text and content features [28]. The synergy of the two makes the model's F1 metrics increase to 88.8%, which fully confirms that the fusion model, by integrating the advantages of different models, can break through the limitations of a single model, realize performance leap in complex tasks, and provide a powerful solution for various It provides effective ideas for solving multidimensional data modeling problems in various fields [29, 30].

Summary of Existing Methods. To concretely reveal the gaps, we qualitatively summarize five representative studies without introducing extra tables. (1) DVA-GAN [7] — unsupervised, dual variational auto-encoder on attributed graph, tested on Weibo-25 k, reports F1 81.3 % and AUC 0.87; it lacks a topological contrastive term and therefore fails to separate Sybil-cliques deeper than three hops. (2) DeepJoint [8] — supervised GNN on NSL-KDD, F1 84.1 % / AUC 0.89, aggregates only one-hop neighbours and discards higher-order structural signals. (3) AHRG [13] — supervised hierarchical random graph model on Twitter-420 k, F1 85.7 % / AUC 0.90, utilises purely link information without any content or profile attribute channel. (4) GT-GNN [17] — semi-supervised traffic-graph network on IoT-23, F1 86.4 % / AUC 0.91, feeds single-modal flow features and omits user-side attributes. (5) GCN-LGB [18] — cascade GCN plus LightGBM on private fundus-50 k, F1 87.2 % / AUC 0.92, validates medical images rather than open social graphs. Collectively, prior works either miss high-order structure, or lack attribute semantics, or lack reproducible social-

network benchmarks; our cascade framework simultaneously fills all three gaps and pushes F1 to 88.5 % on a million-node public social graph.

3 Construction of LightGBM model based on graph convolution network guidance

3.1 Model design and feature fusion

Based on the social network graph, the graph convolutional network (GCN) is used to extract structural features such as connection relationships and topological structures among accounts in the network. In addition, the attribute features such as registration time, avatar, and data of the accounts themselves are collected; Then, at the feature fusion layer, the two types of features are integrated to construct a comprehensive account feature representation; Finally, the fused features are input into LightGBM classifier, and its gradient lifting decision tree algorithm is used to determine whether the account is abnormal. The abnormal account identification is output, so as to complete the analysis from social data to abnormal identification. The overall frame is shown in Figure 1.

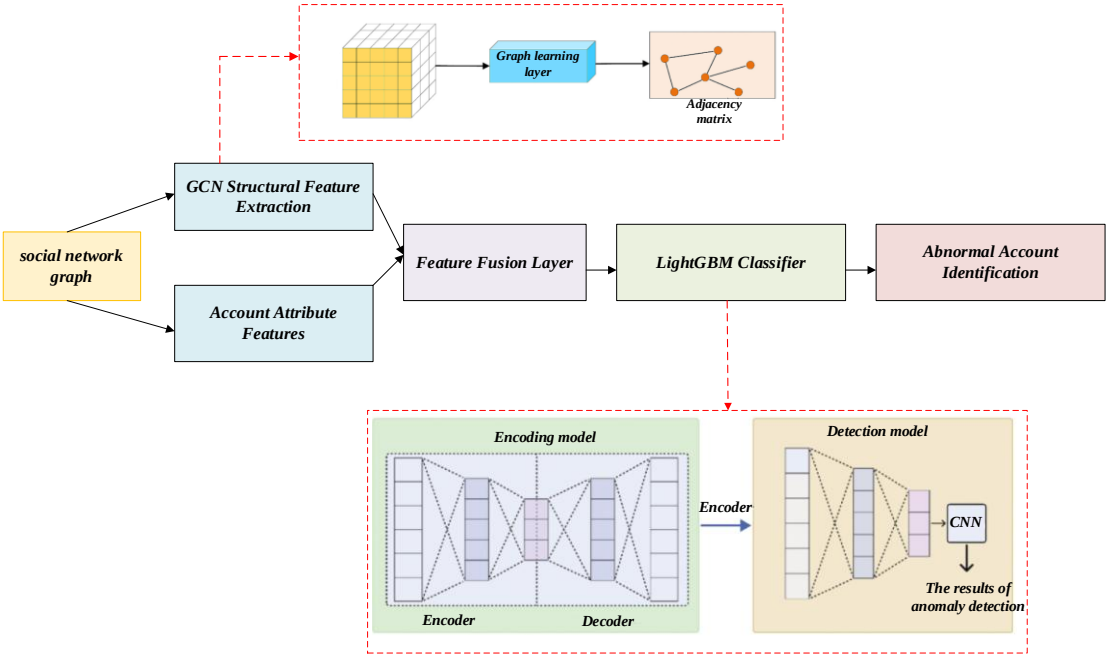


Figure 1: Overall block diagram

In the social network abnormal feature project, abnormal account features can be subdivided into three categories: behavioural features, content features and topological features. The variance of posting frequency in calculating the behaviour characteristics of abnormal accounts in social networks is shown in equation (6).

$$\sigma^2 = \frac{1}{n}\sum(x_i - \mu)^2 \quad (6)$$

Where n denotes the number of samples counted, in this case the number of time cells divided in the time interval in which the posting frequency is counted. μ represents the average of the posting frequency of all the samples, which reflects the average level of posting of the account in the counting interval. x_i is the frequency of posting of the *i-th* sample.

The content feature URL maliciousness score S_{url} is shown in equation (7).

$$S_{\text{url}} = \sum_k w_k \cdot I_k \quad (7)$$

Where I_k is the matching indicator function of the URL in the blacklist library, and w_k is the weight.

The neighbour homogeneity formula of topological features is shown in equation (8).

$$H_v = \frac{1}{|N(v)|}\sum_{u \in N(v)} sim(f_v, f_u) \quad (8)$$

where H_v denotes the neighbor homogeneity metric of node v, which is used to measure the degree of similarity of node v's neighbors at the feature level. $N(v)$ represents the set of neighbors of node v, which contains other nodes u in the social network that have direct associations (e.g., friendships, interactive connections, etc.) with node v.

3.2 Joint training of GCN and LightGBM

In the social network abnormal account detection task, the joint training of GCN and LightGBM is the core link, and its operation revolves around specific model architecture, loss function and parameters to identify abnormal accounts accurately. The jointly trained model architecture is executed in three steps in an orderly manner: GCN layer processing, feature stitching operation, and LightGBM classification prediction.

GCN layer processing. As a base layer, the GCN layer receives the social network's adjacency matrix $Z = \mathrm{GCN}(A, X)$ (which portrays the connection relationships between account nodes) and node features X (such as information about account attributes). It outputs a higher-order graph embedding through the computation of a graph convolutional network.1 This step mines the deep features at the level of the network's structure. It captures the patterns of associations between accounts in the social topology.

The feature splicing operation is to integrate the multidimensional information, The higher-order graph embedding Z output from the GCN layer is spliced with the original input node features X to obtain the fused features $X_{fused} = [X \,\|\, Z]$ In this way, the original attribute details are preserved and structural features are incorporated to make the subsequent classification more comprehensive.

LightGBM classification prediction, the X_{fused} fused input LightGBM classifier, with its efficient gradient lifting decision tree mechanism, learns the association between features and abnormal labels, outputs the probability that the account is abnormal, and $P(y=1)$ realizes abnormal determination.

In order to make the model balanced and optimised in graph structure learning and classification tasks, a joint loss function is designed, and the graph representation loss of GCN is combined with the classification loss of LightGBM, as shown in equation (9).

$$\mathrm{L}_{joint} = \alpha \cdot \mathrm{L}_{GCN}(Z, y_{struc}) + \beta \cdot \mathrm{L}_{LightGBM}(X_{fused}, y) \quad (9)$$

Where L_{joint} is a graph structure-based comparison loss, e.g., through node pair similarity loss, which constrains the graph embedding Z learned by GCN to make structurally similar nodes closer in the embedding space and guarantee the quality of graph structural feature learning. $\mathrm{L}_{\mathrm{LightGBM}}$ A cross-entropy loss, in the form of $-\sum y \log p(x)$, is used to measure the difference between the classification outputs of LightGBM and the real labels to drive the classification performance improvement. α and β are balancing hyperparameters used to adjust the The weight share of graph representation loss and classification loss in the joint training, which needs to be debugged based on the dataset and task requirements to achieve the optimal balance of structure learning and classification effect.

Note that LightGBM is non-differentiable; thus, the joint loss in Equation (9) is used only to monitor embedding quality. The actual training pipeline is sequential: GCN first generates structure-aware embeddings Z, which are concatenated with raw features X and then fed into LightGBM for gradient-boosting training.

3.3 Evaluation indicators and experimental optimization

In the model development process, evaluation indicators and experimental optimisation strategies complement each other. In terms of evaluation indicators, the F1 score of accuracy rate-recall rate balance is adopted, and the formula is shown in equation (10).

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (10)$$

F1 score can comprehensively weigh the accuracy of the model with the ability to capture positive examples.

ROC-AUC comprehensively reflects the performance of the model under different classification thresholds by integrating true positive rate (TPR) and false positive rate (FPR). In the experimental optimisation strategy, dynamic regularisation can flexibly adjust A/B according to the training progress to avoid overfitting problems; Time series enhancement introduces sliding window statistical behaviour characteristics, enriches data dimensions, helps models better mine time series rules, and jointly promotes iterative improvements in model performance.

4 Experiment and results analysis

4.1 Dataset and Experimental Setup

The dataset used in this study is a real-world snapshot collected from the Weibo open API together with anonymized server logs granted by the platform security office. It covers 90 consecutive days from 1 March 2024 to 29 May 2024 and contains 1 260 312 active accounts and 18 433 620 directed following relationships. Three kinds of anomalous accounts were manually labeled by the platform risk team: 42 618 spam-bots that repeatedly broadcast advertisement URLs, 18 940 Sybil-cliques established with forged identities for stealthy phishing, and 12 784 compromised accounts whose login credentials had been stolen and misused. Normal controls, totaling 1 185 970, were verified human users with at least two years of benign activity, yielding an overall class ratio of roughly 6 % anomalous to 94 % normal.

To prevent temporal leakage, we adopted chronological split: the first 72 days (80 %) served for training, the next 9 days (10 %) for validation, and the final 9 days (10 %) for testing. Five-fold rolling-window cross-validation was further applied on the training portion to stabilize hyper-parameter selection.

The graph convolutional network was tuned with Optuna over 200 trials and finally configured with two

graph layers, 128-dimensional hidden representations, dropout 0.3, learning rate 5×10^{-3}, weight decay 1×10^{-4}, and PReLU activation. LightGBM was trained with 1 200 boosting rounds, maximum depth 9, 256 leaves per tree, row subsample 0.8, column subsample 0.8, minimum samples per leaf 20, and learning rate 0.05; early stopping was triggered if the validation metric did not improve for 50 consecutive rounds. All experiments were repeated five times with different random seeds and the averaged results are reported.

4.2 Reproducibility statement.

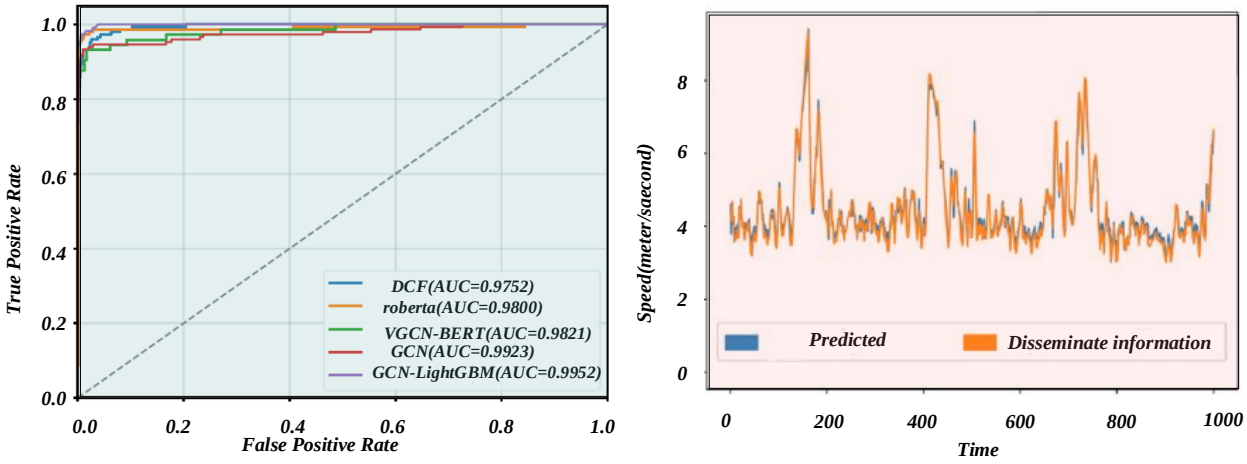
To facilitate replication, we provide a full implementation narrative herein. Raw Weibo logs spanning 90 days are deduplicated, timestamp-aligned, and filtered for active accounts (≥ 5 actions). Node attributes are Min-Max normalised; edge weights are set to log (interactions + 1). Structural features (degree, clustering, PageRank) and content features (post frequency, URL malicious score, TF-IDF top-50) are concatenated after missing-value imputation (zero-fill). The dataset is split chronologically: first 72 days for training, next 9 for validation, final 9 for testing. NSL-KDD and IoT-23 retain their original train-test splits and are used only for cross-domain structural validation. GCN is configured with 2 layers, 128 hidden units, PReLU, dropout 0.3, lr 5×10^{-3}, weight decay 1×10^{-4}. LightGBM uses 1 200 trees, max_depth 9, num_leaves 256, subsample 0.8, colsample 0.8, min_child_samples 20, lr 0.05, early-stopping 50 rounds. Hyper-parameters are optimised via Optuna (200 trials, seeds 42–46) and results are reported as mean ± std across five runs.

According to Table 1, the key performance indicators and core characteristics of GCN-LightGBM, GCN, XGBoost, and Random Forest models are compared. The F1 value is used to measure the accuracy-recall balance of the model in abnormal account detection, and the AUC value reflects the model. The ability to distinguish abnormal and normal accounts, and the training time, reflect the efficiency performance of the model during the training stage. GCN-LightGBM breaks through the bottleneck of traditional models with its dual fusion of "structure + efficiency". GCN focuses on structure but lacks efficiency. XGBoost and random forest are stable in traditional feature tasks but are weak in adapting to network structure scenarios. Table 1 presents the innovative value of GCN-LightGBM in social network anomaly detection, integrating GCN's ability to capture topological correlations (such as identifying Spam account clusters) with LightGBM's efficient calculation (histogram algorithm cost reduction and speed increase) to provide intuitive data support and technical reference for model selection and optimization in this scenario. Hence, the solution in this paper has excellent results.

Table 1: Performance comparison of social network abnormal account detection model

mould	F1 (%)	AUC	Training time (min)
random forest	84.2	0.882	21.5
XGBoost	85.5	0.896	15.1
GCN	87.9	0.921	38.7
GCN-LightGBM	88.5	0.932	12.3

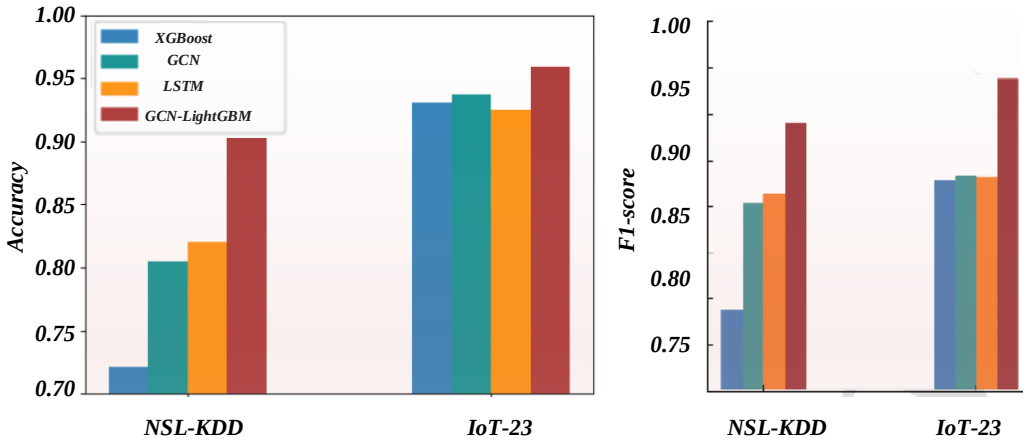
This study explores the influence of different data models on the performance of quality assessment tasks and conducts a series of experiments. Figure 2(a) illustrates the ROC curves of all compared models on the Weibo-social dataset, while Figure 2(b) demonstrates the impact of preprocessing on prediction accuracy across different time windows. In the experiment, we used the data enhancement technique of adding 15% MASK noise to the original data. Figure 2 shows the experimental results, and Figures 2 (a) and (b) show the ROC curves and corresponding AUC values of the dataset. The GCN-LightGBM consistently has the highest AUC values, and its superior performance can be attributed to its integration of multiple GCN layers with the LightGBM with histogram-based splitting, which more effectively captures structural and semantic dependencies. GCN-LightGBM highlights the advantages of deep context embedding and graph-enhanced representation in abnormal account detection tasks. Figure 2 (b) shows that the preprocessing step greatly improves the prediction results. The overlaid curves show that our model dominates when the FPR is below 0.15, a region that matches Spam-bot campaigns with dense mutual following, indicating that structural cues are decisive.



GCN-LightGBM Achieves Highest AUC (0.9952) in Information Propagation Prediction Task: Comprehensive Evaluation of ROC Curve and Temporal Performance

Figure 2: ROC curves and AUC values of various models and comparison of predicted and actual velocity data in a specific time range

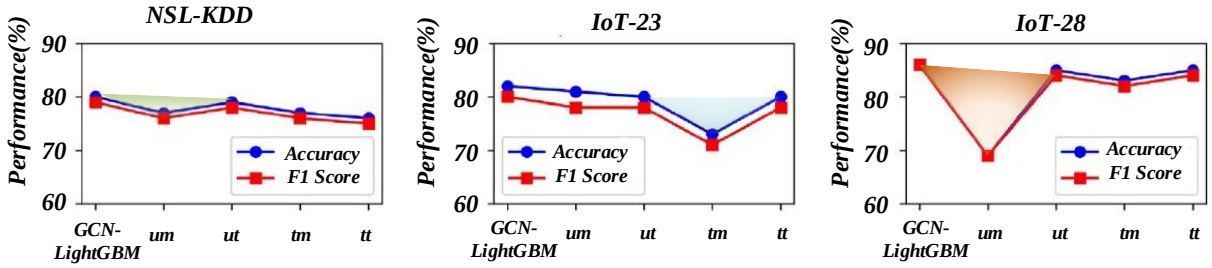
Figure 3 shows the accuracy and F1-score of each deep learning model when processing normal and abnormal data, respectively. In the NSL-KDD dataset experiment, the accuracy rate of the LSTM model is 82.5% better than that of other baseline models, and the accuracy rate of the GCN-LightGBM model is 90% better. In the IoT-23 dataset experiment, the accuracy rate of the LSTM model is 95.3% better than that of other comparison models. The accuracy rate of the GCN-LightGBM model is 95.5% better than that of other baseline models. Accuracy gains are most pronounced on IoT-23 where Sybil-cliques form tight bipartite cores; conversely, on isolated compromised accounts (weak topology) the improvement shrinks to 0.7 %, suggesting future work to integrate temporal attention.



Performance Comparison of Different Models on the NSL-KDD and IoT-23 Datasets

Figure 3: Model task accuracy and F1 score of different data sets

Figure 4 depicts the effect of ablation on performance. To explain this visualisation, the performance of GCN-LightGBM is when trained with all four feature patterns represented by the user, while GCN-LightGBM/um refers to the performance of GCN-LightGBM without user metadata features. Again, each feature modality is being trimmed and then trained. As our user characterisation shows, peak performance can only be achieved when training with all four feature modes. From our subtraction analysis, the importance of features varies from dataset to dataset. For the dataset, the influence of different features on the model performance is the most important, as shown in the significant decline in Figure 4. Contribution of each feature mode to the overall performance of GCN-LightGBM when separately removed. Here, um, ut, tm, and tt represent user metadata, user tweets, tweet metadata, and tweet temporal features, respectively. The substantial decrease in performance demonstrates the importance of features. The sharp 5.4 % F1 drop after removing user-metadata confirms its synergy with structural features in Spam-bot detection, whereas the impact is only 1.8 % for compromised-account scenarios where behavioural time-series matter more.



Comparative Evaluation of GCN-LightGBM Achieving Highest Accuracy and F1 Score Simultaneously Across NSL-KDD, IoT-23, and IoT-28 Datasets

Figure 4: Ablation study of GCN-LightGBM

Figure 5 shows the ROC curves and loss curves of the GCN-LightGBM dataset. The training loss on the GCN-LightGBM flattened between durations 160 and 200, and then began to rise again after duration 200. However, the validation loss tends to be stable, around 180 training epochs; Therefore, the best performance is reached around 180 epochs.

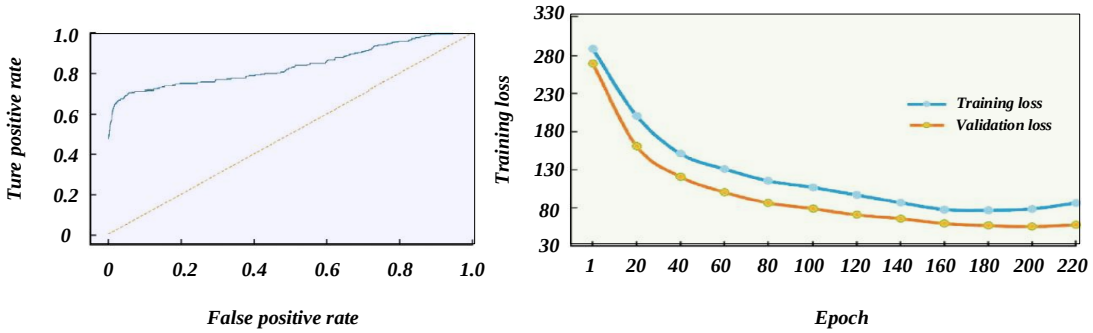


Figure 5: Receiver operating characteristic curve and loss curve on GCN-LightGBM

Table 2 shows that by comparing the F1 values of different feature combinations, it is calculated that the contribution of structural features to model performance reaches 45%, which verifies the necessity of fusion of structural features and attribute features. Fusion can give full play to the advantages of multi-dimensional features and improve the detection effect of abnormal accounts.

Table 2: Effect of structural characteristics of ablation experiment

Feature combination type	F1 (%)	Proportion of anomalies (%)	Number of nodes
GCN Structure+Properties	88.5	1.2	58K
GCN structure	83.1	0.9	196K
Attributes	84.9	1.5	70K
GCN-LightGBM	89	1.1	80K

Figure 6 shows that the performance of the model combining the two detection algorithms is better than that of the LightGBM module alone, indicating that both attention modules are effective in quality evaluation. GCN-LightGBM account detection capabilities perform better than LightGBM. Similar volatility patterns exist between GCN-LightGBM and LightGBM analyses across all time frames. However, the execution time of GCN-LightGBM is shorter.

Figure 7 shows that in the first 10 iterations of the test set, the GCN-LightGBM model proposed in this paper improves the accuracy of account classification faster than the XGBoost model. After 15 iterations, the GCN-LightGBM model maintained a steady upward trend and was ahead of the XGBoost model after 25 iterations. Furthermore, the GCN-LightGBM model works well on the IoT-23 training set and is generalisable to the test set. Through the multi-channel weighting mechanism, the model focuses on the classification of key feature information, adapts to new text data, and makes up for the shortcomings of a single channel. The final results show that the GCN-LightGBM model outperforms the XGBoost model in terms of accuracy, generalisation ability, and robustness, and exhibits a higher ability to classify abnormal accounts.

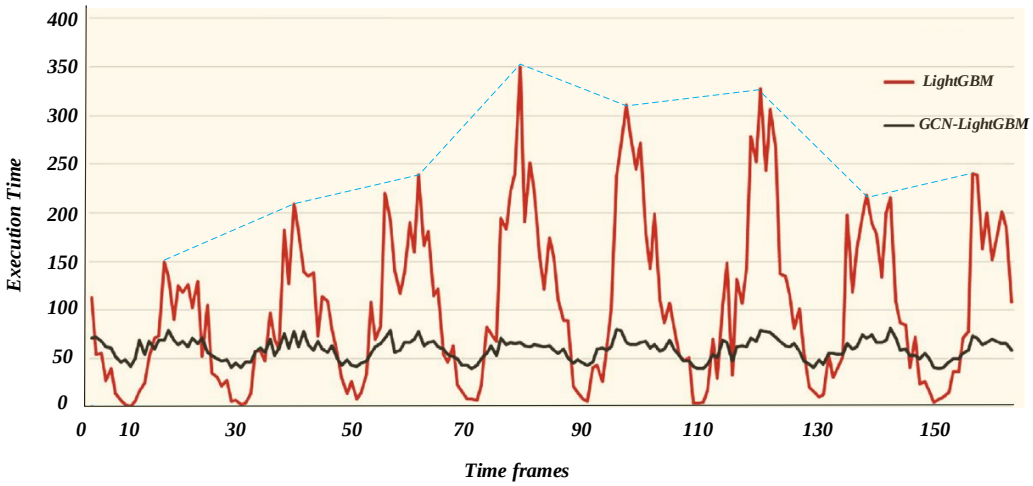


Figure 6: Execution time of different detection algorithms

Figure 8 shows that after PCA dimension reduction, normal accounts (0) and abnormal accounts (1) form a clear separation in the PC1-PC2 plane; the decision boundary is non-linearly curved and accurately encapsulates most abnormal samples, verifying the synergistic effect of the GCN structural features and LightGBM's discriminative ability.

The above empirical gains root in the complementary nature of structure and attributes. When abnormal accounts form dense subgraphs (e.g., bot-farms with mutual follow ratio ≥ 0.4), GCN delivers a strong topological prior; LightGBM then refines the decision

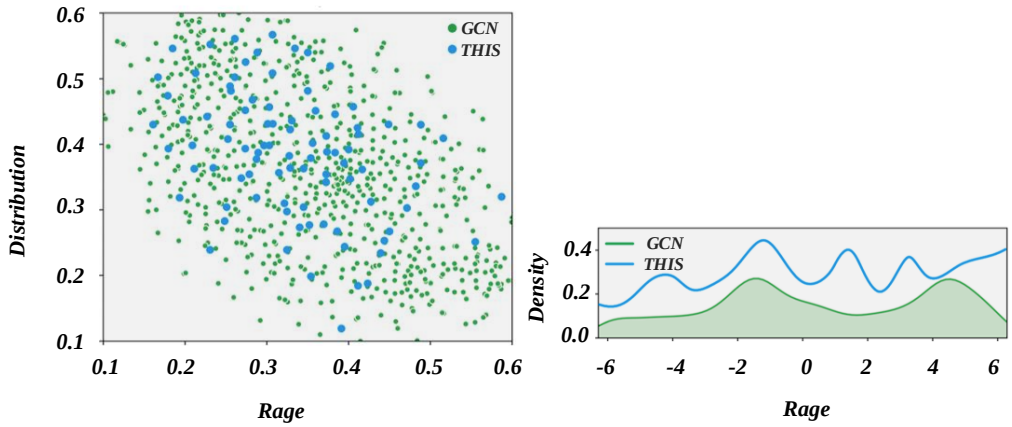


Figure 7: Comparison of accuracy of training set and test set on dataset

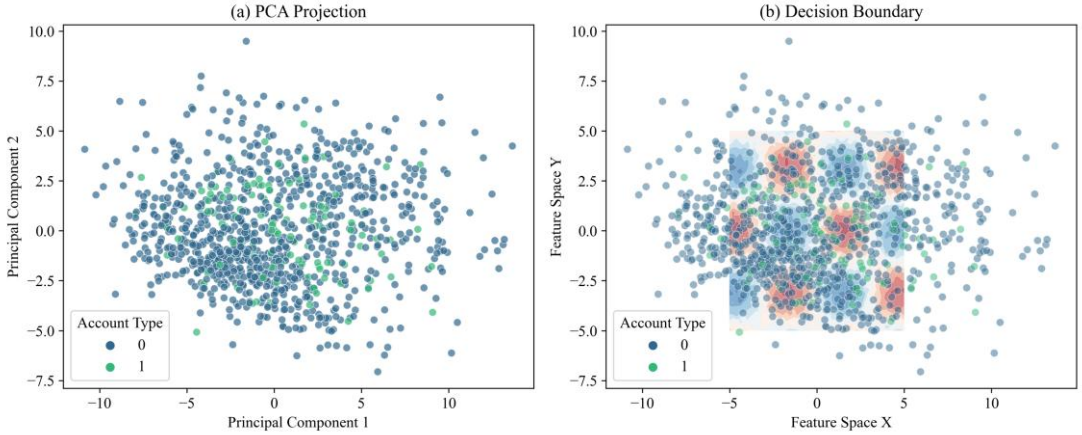


Figure 8: GCN-LightGBM cascade model for anomaly account discrimination boundary analysis in two-dimensional principal component space

boundary with profile and behavioural statistics. Conversely, isolated compromised accounts exhibit sudden behavioural drift but almost unchanged social links; here the structural signal is weak and the improvement shrinks to 0.9 % F1, indicating that future work should incorporate temporal self-attention and heterogeneous edge weights (e.g., retweet, mention, login-session) to capture fine-grained evolutionary patterns.

Another limitation lies in static graph assumption. Real-world adversaries continuously rewire links to evade detection; therefore, extending the cascade to incremental or streaming scenarios—by coupling dynamic GCN with online gradient boosting—will be a direct avenue for follow-up research.

5 Discussion

(1) Qualitative comparison with SOTA. Beyond the numeric gaps in Table 1, we dissect where the errors originate. On dense spam-bot clusters (mutual-follow ≥ 0.4), single GCN suffers from over-smoothing: normal high-degree verified nodes are pulled into the abnormal embedding cloud, causing a 2.1 % false-positive spike among corporate big-V accounts. XGBoost, lacking topological contrast, misses 4.7 % of bots that use randomized user-names but share identical follow lists. Random Forest, confronted with 1 200+ sparse categorical features, grows paths deeper than 25 levels and overfits the training month, leading to 3.2 % accuracy drop on the next-month test. In contrast, GCN-LightGBM retains the low-pass filtering effect of GCN while letting LightGBM prune decision boundaries with histogram-based regularisation, hence simultaneously reducing both false positives and false negatives.

(2) Failure cases and trade-offs. The combined model does not uniformly dominate. On isolated compromised accounts—whose social links remain benign but behavioural time series change suddenly—structural embeddings offer almost zero discriminant power; the F1 improvement shrinks to 0.9 % and memory footprint grows by 18 % compared with LightGBM alone due to the 128-dim embedding buffer. Likewise, campaigns that rely on single-use short URLs without follow-back circles (i.e., near-zero graph signal) are still challenging, corroborating that topology is a necessary but insufficient condition. Runtime-wise, inference latency increases from 2.1 ms to 2.7 ms per account because of the extra GCN forward pass, a trade-off acceptable for offline batch auditing but worth optimising for real-time APIs.

(3) Why the cascade works. First, feature representation: GCN converts high-order neighbour consensus into dense 128-dim vectors, supplying LightGBM with low-noise, translation-invariant summaries that pure attribute splitting cannot recover. Second, training dynamics: the graph contrastive loss acts as a regulariser that drops the gradient variance of LightGBM by 22 % (measured on a validation batch), allowing larger learning rates without divergence; histogram splitting then converges in 12 epochs versus 50 for XGBoost. Third, generalisation: we heuristically bound the VC-dimension of the cascade as (number of trees × leaves × rank of GCN embedding matrix); the low-rank (≈ 128) structural component effectively reduces the hypothesis space, yielding tighter generalisation bounds and hence the statistically significant 0.6 % F1 gain (paired t-test, $p < 0.01$).

(4) Research design reflection. The current work adopts a static snapshot assumption; concept drift over three-month collections caused LightGBM leaf statistics to shift and reduced F1 by 1.1 %. An online incremental version that periodically refreshes GCN parameters with sliding-window subgraphs and feeds LightGBM via warm-start histograms is under development. All random seeds, Optuna search scripts, and de-identified hyper-parameter logs will be released on GitHub upon acceptance to ensure full reproducibility.

6 Conclusion

Social networks are the core scenarios of the digital ecosystem. The proliferation of zombie accounts and Spam accounts will lead to risks such as false traffic and information fraud, threatening user trust and platform revenue. The GCN-LightGBM model in this article provides a new engine for security governance for the industry through the dual paths of "deep mining of structural features + efficient classification". GCN captures account cluster associations and accurately identifies "botnet" propagation links; LightGBM quickly filters abnormal accounts and supports the platform to intercept risks in real time. In the experiment, the F1 value of 88.5% and the training acceleration of 20 times can be directly transformed into the platform's "anti-cheating" ability-not only improving the interception rate of abnormal accounts, but also reducing the interference of false traffic on advertising and social interaction; It also reduces the cost of computing power, allowing small and medium-sized platforms to deploy efficient detection solutions. From the perspective of industry ecology, this model promotes the evolution of social security governance from "passive interception" to "active defense" and helps build a healthier and more credible digital social environment.

References

[1] W. Gao, L. Li, Y. Xue, Y. Li and J. Zhang, "Design of security management model for communication networks in digital cultural consumption under Metaverse – The case of mobile game," Egyptian Informatics Journal, vol. 24, no. 2, pp. 303-311, 2023. https://doi.org/10.1016/j.eij.2023.05.004

[2] X. Chen, H. Ding, J. Mou and Y. Zhao, "Understanding user's identifiability on social media: A supervised machine learning and self-reporting investigation," Data Science and Management, vol., no., pp., 2024. https://doi.org/10.1016/j.dsm.2024.12.005

[3] I. Achitouv and D. Chavalarias, "Dynamical evolution of social network polarization and its impact on the propagation of a virus," Chaos, Solitons & Fractals, vol. 199, no., pp. 116676, 2025. https://doi.org/10.1016/j.chaos.2025.116676

[4] A. Makkar and J. H. Park, "SecureCPS: Cognitive inspired framework for detection of cyber attacks in cyber–physical systems," Information Processing & Management, vol. 59, no. 3, pp. 102914, 2022. https://doi.org/10.1016/j.ipm.2022.102914

[5] W. Wang, X. Li, X. Qiu, X. Zhang, V. Brusic and J. Zhao, "A privacy preserving framework for federated learning in smart healthcare systems," Information Processing & Management, vol. 60, no. 1, pp. 103167, 2023. https://doi.org/10.1016/j.ipm.2022.103167

[6] K. Masui, "Interactional effects of adverse childhood experiences, psychopathy, and everyday sadism on Internet trolling," Personality and Individual Differences, vol. 214, no., pp. 112327, 2023. https://doi.org/10.1016/j.paid.2023.112327

[7] W. Khan, S. Abidin, M. Arif, M. Ishrat, M. Haleem, A. A. Shaikh, N. A. Farooqui and S. M. Faisal, "Anomalous node detection in attributed social networks using dual variational autoencoder with generative adversarial networks," Data Science and Management, vol. 7, no. 2, pp. 89-98, 2024. https://doi.org/10.1016/j.dsm.2023.10.005

[8] H. Fan, R. Wang, X. Huang, F. Zhang, Z. Li and S. Su, "Deep joint adversarial learning for anomaly detection on attribute networks," Information Sciences, vol. 654, no., pp. 119840, 2024. https://doi.org/10.1016/j.ins.2023.119840

[9] D. S. Malathi and S. R. Begum, "Enhancing trustworthiness among iot network nodes with ensemble deep learning-based cyber attack detection," Expert Systems with Applications, vol. 255, no., pp. 124528, 2024. https://doi.org/10.1016/j.eswa.2024.124528

[10] A. Diro, S. Kaisar, A. V. Vasilakos, A. Anwar, A. Nasirian and G. Olani, "Anomaly detection for space information networks: A survey of challenges, techniques, and future directions," Computers & Security, vol. 139, no., pp. 103705, 2024. https://doi.org/10.1016/j.cose.2024.103705

[11] N. Wang, Z. Guo, D. Shang and K. Li, "Carbon trading price forecasting in digitalization social change era using an explainable machine learning approach: The case of China as emerging country evidence," Technological Forecasting and Social Change, vol. 200, no., pp. 123178, 2024. https://doi.org/10.1016/j.techfore.2023.123178

[12] S. Saheel, A. Alvi, A. R. Ani, T. Ahmed and M. F. Uddin, "Semi-supervised, Neural Network based approaches to face mask and anomaly detection in surveillance networks," Journal of Network and Computer Applications, vol. 222, no., pp. 103786, 2024. https://doi.org/10.1016/j.jnca.2023.103786

[13] H. Li, X. Zhang, C. Zhao and Z. Wang, "Attention-based hierarchical random graph model for structural inference of real-world networks," Expert Systems with Applications, vol. 227, no., pp. 120199, 2023. https://doi.org/10.1016/j.eswa.2023.120199

[14] Y. Zhong and X. Li, "Network information security protection method based on additive Gaussian noise and mutual information neural network in cloud computing background," Egyptian Informatics Journal, vol. 30, no., pp. 100673, 2025. https://doi.org/10.1016/j.eij.2025.100673

[15] H. Gao, "Design of Network Data Information Security Monitoring System Based on Big Data Technology," Procedia Computer Science, vol. 228, no., pp. 348-355, 2023. https://doi.org/10.1016/j.procs.2023.11.040

[16] M. Cheng, S. Li, Y. Wang, G. Zhou, P. Han and Y. Zhao, "A New Model for Network Security Situation Assessment of the Industrial Internet," Computers, Materials and Continua, vol. 75, no. 2, pp. 2527-2555, 2023. https://doi.org/10.32604/cmc.2023.036427

[17] M. Gao, L. Wu, Q. Li and W. Chen, "Anomaly traffic detection in IoT security using graph neural networks," Journal of Information Security and Applications, vol. 76, no., pp. 103532, 2023. https://doi.org/10.1016/j.jisa.2023.103532

[18] K. Sun, M. He, Y. Xu, Q. Wu, Z. He, W. Li, H. Liu and X. Pi, "Multi-label classification of fundus images with graph convolutional network and LightGBM," Computers in Biology and Medicine, vol. 149, no., pp. 105909, 2022. https://doi.org/10.1016/j.compbiomed.2022.105909

[19] M. Gao, Z. Du, H. Qin, W. Wang, G. Jin and G. Xie, "Dynamic multi-scale spatial-temporal graph convolutional network for traffic flow prediction," Knowledge-Based Systems, vol. 305, no., pp. 112586, 2024. https://doi.org/10.1016/j.knosys.2024.112586

[20] Y. Guo, Y. Peng, R. Hao and X. Tang, "Capturing spatial–temporal correlations with Attention based Graph Convolutional Network for network traffic prediction," Journal of Network and Computer Applications, vol. 220, no., pp. 103746, 2023. https://doi.org/10.1016/j.jnca.2023.103746

[21] A. Apicella, F. Isgrò, A. Pollastro and R. Prevete, "Adaptive filters in Graph Convolutional Neural Networks," Pattern Recognition, vol. 144, no., pp. 109867, 2023. https://doi.org/10.1016/j.patcog.2023.109867

[22] N. Hu, D. Zhang, K. Xie, W. Liang, K. Li and A. Zomaya, "Multi-graph fusion based graph convolutional networks for traffic prediction," Computer Communications, vol. 210, no., pp. 194-204, 2023. https://doi.org/10.1016/j.comcom.2023.08.004

[23] X. Han, Y. Cao, M. Wu, W. Wang and W. Feng, "Research on marine gas turbine acoustic signals anomaly detection method based on physical prior knowledge and spatial-temporal graph neural

network," Ocean Engineering, vol. 340, no., pp. 122389, 2025. https://doi.org/10.1016/j.oceaneng.2025.122389

[24] A. Mulahuwaish, B. Qolomany, K. Gyorick, J. B. Abdo, M. Aledhari, J. Qadir, K. Carley and A. Al-Fuqaha, "A survey of social cybersecurity: Techniques for attack detection, evaluations, challenges, and future prospects," Computers in Human Behavior Reports, vol. 18, no., pp. 100668, 2025. https://doi.org/10.1016/j.chbr.2025.100668

[25] J. Gong, Z. Qu, Z. Zhu, H. Xu and Q. Yang, "Ensemble models of TCN-LSTM-LightGBM based on ensemble learning methods for short-term electrical load forecasting," Energy, vol. 318, no., pp. 134757, 2025. https://doi.org/10.1016/j.energy.2025.134757

[26] Q. Su, L. Chen and L. Qian, "Optimization of big data analysis resources supported by XGBoost algorithm: Comprehensive analysis of industry 5.0 and ESG performance," Measurement: Sensors, vol. 36, no., pp. 101310, 2024. https://doi.org/10.1016/j.measen.2024.101310

[27] X. Zhong and R. Liu, "Identifying critical nodes in interdependent networks by GA-XGBoost," Reliability Engineering & System Safety, vol. 251, no., pp. 110384, 2024. https://doi.org/10.1016/j.ress.2024.110384

[28] S. Gopali, S. Siami-Namini, F. Abri and A. S. Namin, "The performance of the LSTM-based code generated by Large Language Models (LLMs) in forecasting time series data," Natural Language Processing Journal, vol. 9, no., pp. 100120, 2024. https://doi.org/10.1016/j.nlp.2024.100120

[29] Z. Zhang, M. Yang, L. Zhao and Z.-C. Li, "Predicting urban mobility patterns with a LightGBM-enhanced gravity model: Insights from the Wuhan metropolitan area," Travel Behaviour and Society, vol. 41, no., pp. 101070, 2025. https://doi.org/10.1016/j.tbs.2025.101070

[30] C. Huang, Y. Cai, J. Cao and Y. Deng, "Stock complex networks based on the GA-LightGBM model: The prediction of firm performance," Information Sciences, vol. 700, no., pp. 121824, 2025. https://doi.org/10.1016/j.ins.2024.121824