Graph Neural Network-Based User Preference Model for Social Network Access Control

Yuan Zhang[1,2*]
[1]Xuchang Vocational Technical College, Xuchang 461000, China
[2]Henan Province Data Intelligence and Security Application Engineering Technology Research Center, Xuchang 461000, China
E-mail: hnxc_z@126.com
[*]Corresponding author

Keywords: social networks, user preferences, graph neural network, multi-layer attention, access control

Received: November 11, 2024

The popularity and deepening of social networks have increased the risk of personal information leakage for users. To enhance the security of social networks, this study constructed an access control model based on the preferences of social network users. This model utilizes graph neural networks to generate access control strategies based on user preferences, and introduces a multi-layer attention mechanism to optimize the graph neural network. To better capture user preference information, the study sets the learning rate to 0.0001. The experimental results demonstrated that in the Twitter dataset, the accuracy of the proposed model reached 95.7% and the F1 score reached 96.2%, which were significantly higher than those of other models. These results indicated that the model could more accurately classify access control in social networks and reduce false positives. The area under the receiver operation characteristic curve of the proposed model was 0.982, which was higher than other models. The decision time was 13.77 seconds, significantly lower than other models. This indicated that the model could more effectively distinguish different types of user access requests and provide more reliable guarantees for secure access to social networks. The user's preferred social network access control model based on graph neural networks has superior performance, effectively ensuring the information security of social network users and laying the foundation for further development of access control technology.

Povzetek: Predstavljen je nov model za nadzor dostopa v družbenih omrežjih, ki temelji na grafskih nevronskih mrežah in uporabniških preferencah. Z uporabo večslojnega pozornostnega mehanizma model omogoča zanesljivo in varno upravljanje dostopa.

1 Introduction

In the era of rapid digital development, social networks play an important role in today's society. Through social platforms, people can not only obtain the information and exchange ideas they need but also engage in commercial activities through social networks, greatly changing their communication methods and lifestyle habits [1, 2]. However, the popularity of social networks has made the issue of user privacy protection increasingly prominent. Using social networks means that users need to expose their personal information to a certain extent. Criminals can steal user information through cyber attacks and use it for illegal activities, thereby posing potential risks to users [3]. Meanwhile, there is a large amount of false information and rumors on social networks. The rapid dissemination of this information may lead to misunderstandings among the public about certain events or issues, resulting in adverse social impacts. Access control is a critical component in information security, used to manage user access permissions to systems, networks, or applications. Access control can help organizations protect important data and resources from unauthorized access and malicious activities. It can also prevent data leakage, tampering, and destruction, protect sensitive information from being leaked to unauthorized personnel, and ensure the reliability of network systems [4]. Therefore, the importance of implementing effective access control for social networks is self-evident. Nowadays, there are mainly attribute-based, policy-based, and relation-based Access Control Models (ACM), which are widely used in various scenarios [5]. However, traditional models still have drawbacks such as complex permission management and difficulty in adapting to dynamic network environments. Specifically, traditional models often require manual intervention in the process of assigning, revoking, and updating permissions, resulting in increased management costs and error rates. The user behavior and social relationships of social networks are constantly changing, and traditional models are difficult to adapt, resulting in insufficient flexibility of access control policies and inability to effectively respond to new security threats. In this context, this study constructs ACM based on the preferences of social users, uses Graph Neural Network

(GNN) for access control, and introduces Multi-Layer Attention (MLA) to optimize GNN. Finally, UP-GNN-SNAC model, a GNN-based social network ACM catering to user preferences, is designed. The innovation of the research lies in constructing an ACM based on user preferences. Compared with existing GNN-based ACMs, this model better balances privacy protection and user experience by capturing user preferences, providing a more efficient and accurate solution for secure access to social networks.

2 Related works

The progress of the Internet has made it a part of people's daily life to interact with others through social networks. However, due to system vulnerabilities in online platforms, many criminals exploit these vulnerabilities to launch attacks, resulting in the leakage of user information and even being maliciously exploited. Access control, as a key technology for maintaining social network security, is currently a hot topic of research among relevant professionals. You M et al. designed a knowledge graph-based access control decision-making method to improve access control performance under different degrees of imbalance. It extracted topological features to represent high cardinality classification users and resource attributes, revealing the interrelationships between different objects. This method could significantly improve access control performance [6]. Gai K et al. designed a zero-trust cross-organizational data sharing ACM based on blockchain to enhance security in network data sharing. It utilized blockchain alliances to establish a trusted environment and deployed role-based access control through multi-signature protocols and smart contract methods, which had high practicality [7]. Wu H et al. designed a cloud network secure storage data ACM based on association rules to improve the security of social network data access control. It utilized association rule feature extraction methods for data mining and attack detection in network security storage areas and achieved data access control in network security storage areas through adaptive partition-weighted interface scheduling. This method was superior to traditional methods [8]. Azbeg K et al. designed an ACM based on improved blockchain technology to enhance the security and privacy of network systems. It stored data in the interstellar file system and utilized authorization proof-based Ethereum access blockchain to accelerate data storage. This method could significantly improve network security [9]. Zhang L et al. designed a lightweight decentralized multi-authorization ACM based on ciphertext policy attribute-based encryption and blockchain to enhance the security of in-vehicle social networks. Distributed multi-authorization nodes supported vehicle users by performing lightweight computing with the help of vehicle cloud service providers. This model had significant advantages compared to existing solutions [10].

Zhao Y et al. designed a policy-protected, cleanable ACM to improve the efficiency of data encryption in vehicle social networks. It could test and clean encrypted data, and divide access policies into attribute names and attribute values, thereby hiding information in the ciphertext and achieving good encryption performance [11]. Squicciarini A et al. designed a discrete ACM based on individual decision-making to address privacy and security issues arising from data sharing in social networks. It took into account individual preferences in social networks and selected discrete privacy values from a fixed set of options. This model had a good privacy protection effect in data sharing [12]. Dixit M S et al. designed a deep learning-based real-time user ACM for social networks to address user login restrictions. It used CNN and LSTM to predict the age of users and adopts multi-task CNN for face detection and feature extraction, thus achieving significant control over user login [13]. Wen W et al. designed an autonomous privacy control and identity verification sharing scheme built on fast response codes in social networks to solve the problem of users being unable to independently control privacy sharing. It used fast response codes with high-quality images for error correction, combining the advantages of polynomial-based and visual-based secret image sharing. This scheme had low computational complexity and scalability [14]. Safi S M et al. designed an improved end-to-end mobile social network security ACM to protect the personal privacy of social network users. It encrypted user-shared data through ciphertext policy attribute encryption, utilizing advanced encryption standards to prevent unauthorized user access. This scheme had high security and practicality [15]. The summary of related work is shown in Table 1.

Table 1: Summary of related work.

References	Model	Key features	Dataset	Indicator results	Insufficient
[6]	Access Control Decision Method Based on Knowledge Graph	Extracting topological features to represent user and resource attributes	Synthesize social network data	Improved access control performance	Not considering the balance between privacy protection and user experience
[7]	Blockchain Zero Trust Cross	Establishing a Trusted	Cross organizational	High practicality	Slow response speed

	Organizational Data Sharing Access Control	Environment through Blockchain Alliance	transaction data		
[8]	Association Rules Cloud Network Security Storage Access Control	Using association rules for attack detection	Cloud storage logs	Superior to traditional methods	Poor adaptability to new types of attack modes and difficulty in handling dynamically changing environments
[9]	Improve blockchain technology access control	Interstellar File System and Authorization Proof Ethereum	File transfer record	Significant improvement in network security	Requires a large amount of storage space
[10]	Decentralized Multi Authorization Model for Vehicle mounted Social Networks	Cryptography policy attribute-based encryption and blockchain	Vehicle communication records	Compared to existing solutions, it has significant advantages	Complex key management increases deployment difficulty and slow response speed
[11]	Policy protection can purify access control	Testing and cleaning encrypted data	Encrypt the dataset	The encryption effect is good	The cleaning process may result in information loss
[12]	Individual Decision Discrete ACM	Personal preference privacy protection in social networks	user behavior data	Good privacy protection effect	Lack of effective modeling of group behavior and insufficient consideration of personalized preferences
[13]	Real time user access control in deep learning	Convolutional neural network predicts age	Social network user data	Superior to traditional methods	Deep learning models require a large amount of data for training, which poses a risk of privacy leakage
[14]	Quick response code autonomous privacy control	Image correction combined with secret image sharing	User uploads images	Low computational complexity and good scalability	High image quality requirements and sensitivity to image noise
[15]	Mobile social network security access control	Cryptography policy attribute encryption	Mobile device logs	High safety and practicality	The key distribution and management of ciphertext policy attribute encryption are relatively complex

In summary, many scholars have achieved significant results in social network access control. However, these methods still have slow response times and fail to consider the balance between privacy protection and user experience. Therefore, this study constructs an ACM based on user preferences and simulates it using an improved GNN with MLA mechanism to design an UP-GNN-SNAC model to improve access control effectiveness.

3 GNN-based ACM based on user preferences

This section mainly elaborates on the construction process of the UP-GNN-SNAC model. The first section is the design of ACM based on user preferences, and the second section is the implementation of an access control algorithm based on improved GNN.

3.1 ACM construction based on user preferences

User preference refers to the preferences of users towards certain things, which are formed by the comprehensive influence of various factors such as personal factors and social environment. Among them, personal factors include internal characteristics such as age, gender, occupation, interests, values, and behavioral habits of users. Social factors include external environmental factors such as social circles, interaction objects, social frequency, cultural values, and social interactions. In social networks, users express their preferences through posting and activity operations. These operations generate a large amount of data. By analyzing these data, user behavior patterns and characteristics can be understood, and appropriate access permissions can be generated for users to meet their privacy needs in different scenarios, thereby protecting user privacy [16]. Therefore, this study constructs a model based on the preferences of social network users, as shown in Figure 1.

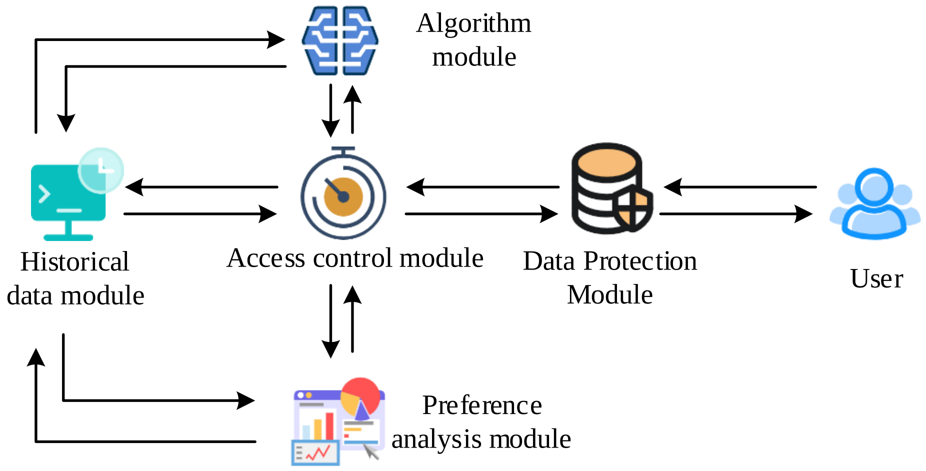


Figure 1: Specific architecture of access control model.

In Figure 1, ACM consists of six modules: user, data protection, access control, preference analysis, historical data, and algorithm. When users need to post or obtain information from social networks, sending requests first goes through the data protection module. It can encrypt and backup the information posted by users. Then, the data protection module sends the user's request to the access control module. This module sends requests to the preference analysis module, historical data module, and algorithm module respectively. The historical data module can extract and preprocess user interaction behavior data, basic attributes, and social relationship data. After cleaning, deduplication, and standardization of these data results, they provide input for the preference analysis module and algorithm module. When the request sent by the data protection module is transmitted to the preference analysis module, it will analyze the user's historical social data, obtain the user's preferences, and return Personal Preferences (PPs). When the data is transmitted to the algorithm module, it will train the obtained data and finally return the best result to the access control module. Specifically, different users have different preferences. When users upload information, different preference information corresponds to different access control policies [17]. It is necessary to determine the level of privacy of uploaded information based on user preferences, that is, to establish a quantitative model of user preferences to measure social information entropy. Figure 2 shows a social information sensitivity measurement model based on user preferences.

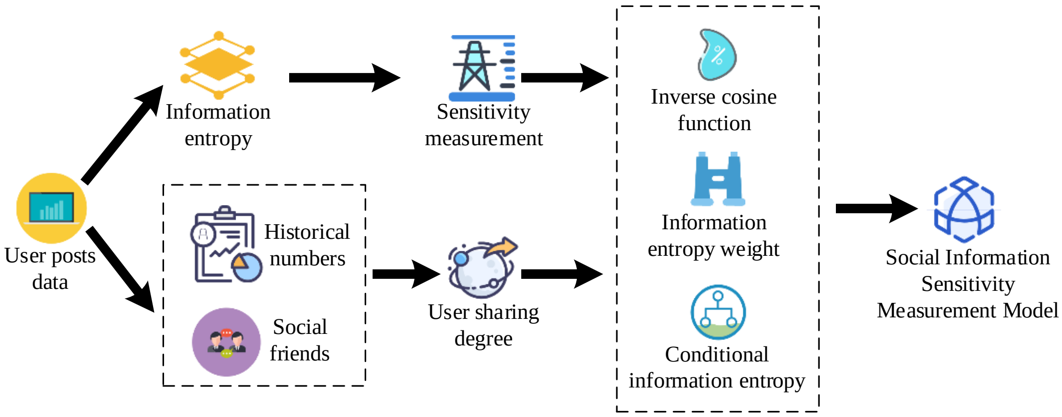


Figure 2: Social information sensitivity measurement model based on user preferences.

In Figure 2, after users post information, they need to calculate the sensitivity of the information based on information entropy and obtain the user's social information sharing degree based on their historical visits and social friends. It is also necessary to use methods such as information entropy weight and conditional information entropy to calculate and obtain an information entropy measurement model. Information entropy is a basic concept in information theory, which is used to measure the uncertainty of a random variable. It reflects how difficult it is to predict the outcome of an event, or how much information is needed to describe the event. Conversely, an increase in information entropy corresponds to a decrease in event outcome uncertainty, and vice versa. Information entropy weight is a weight allocation method based on information entropy, which is used to measure the importance of different features or data dimensions. Conditional information entropy is used to measure the uncertainty of a random event given certain conditions. Therefore, information entropy can be used to describe the amount of privacy contained in social data, determine the degree of privacy of the social data, and construct an information sensitivity measurement model for social data. The calculation method for the amount of social data of users is shown in equation (1).

$$H(x) = -\sum p(x_i) log_2 p(x_i) \, (1)$$

In equation (1), $H(x)$ is the average number of private information uploaded by all users in the social network. $p(x_i)$ is the proportion of the privacy level of information i in the total privacy information.

As the social breadth of a user increases with the number of social friends, the relationship between the social breadth of a user and the number of social friends can be obtained as shown in equation (2).

$$w(F) = \frac{2}{\pi} \arctan F \, (2)$$

In equation (2), $w(F)$ represents the social breadth of the user. F is the number of social friends of the user. The confidentiality of the information posted by the user can be calculated based on whether the user's uploaded information is blocked from their friends. The calculation method is shown in equation (3).

$$h_i = \frac{F_a^i}{F} \, (3)$$

In equation (3), h_i is the confidentiality level of information i. F_a^i is the number of friends blocked by the user.

The level of confidentiality is the ratio of the number of friends allowed to view social data on a social network to the total number of friends. As the number of friends permitted to view the social data increases, the level of confidentiality thereof decreases. The degree of social information sharing can be used to describe the impact of the number of social friends and the number of friends blocked by the user on social data sharing. The degree to which friends are permitted to access information is directly correlated with the extent of social information sharing. The calculation method is shown in equation (4).

$$s_i(F, F_a^i) = h_i \times w(F) = \frac{2F_a^i \arctan F}{\pi F} \, (4)$$

In equation (4), $s_i(F, F_a^i)$ is the user's social information sharing degree. Information entropy can be measured based on the degree of social information sharing among users, as shown in equation (5).

$$H_s(x) = -\sum s_i p(x_i) \log_2 p(x_i) \, (5)$$

In equation (5), $H_s(x)$ is the information entropy. According to the information entropy analysis of user preference mechanism, in the algorithm module, user social data are divided into a training set and a testing set, and they are trained separately to obtain the final access control policy. Figure 3 displays the obtaining process of the strategy.

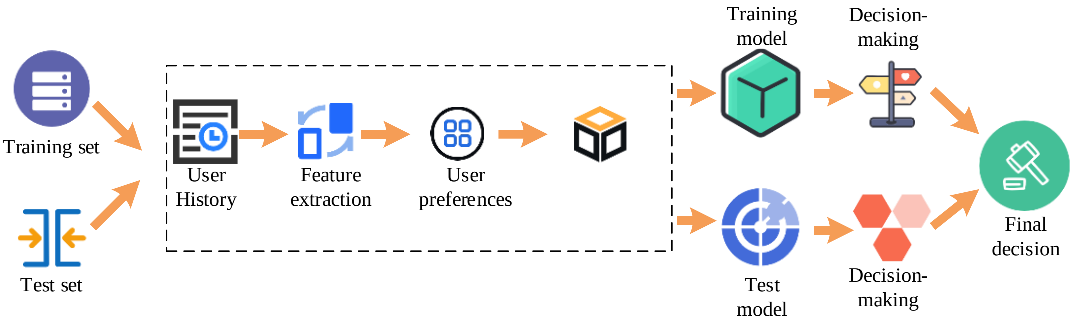


Figure 3: Access control policy acquisition process.

In Figure 3, after dividing the social data into two datasets, the user history records of different datasets are first obtained, and then user feature extraction is performed to calculate user preferences. The next step is to combine the feature vectors to obtain a training model and a testing model, and then make decisions separately through the training model and the testing model. The final step is to make a comprehensive decision based on the access request, obtain the final decision, and thus obtain the access control policy.

3.2 Access control method based on improved GNN

In ACM, the algorithm module is the core of the entire model, which is used to process and analyze the dynamic processing unit of user preference data, historical behavior data, and social relationship data. The algorithm module not only determines whether the model can accurately analyze user preferences but also directly relates to whether the model can make effective decisions [18]. GNN is a graph-based deep learning method that can enrich node representations by utilizing the relationships between nodes. Specifically, GNN can update the representation of nodes by defining the connection relationships between nodes on the graph, utilizing their neighbor information to achieve information transfer and learning of the entire graph. GNN mainly includes three core functions: node representation, graph structure representation, and message passing. Among them, node representation can map each node to a low dimensional vector space for subsequent calculations. The graph structure is represented in a low dimensional vector space for subsequent calculations. Message passing is defined as the process by which a node updates its own representation by exchanging information with its neighboring nodes. This process enables the transmission of information on the graph [19]. Attention mechanism is an important technique in deep learning that allows models to selectively focus on different parts of the input sequence, assigning different weights to each part of the input sequence to highlight the more critical information for the task. The attention mechanism is a process that dynamically assigns weights to the elements of the input sequence. This allows the model to focus on key parts of the input in a targeted manner. As a result, the model processes and learns information in the data more efficiently [20]. Therefore, GNN performs well in graph structure data such as social networks and chemical molecular structures. Research is conducted on constructing a GNN model based on user preferences. To improve the performance of the model, MLA mechanism is introduced to optimize the model, and an access control method based on improved GNN is designed to capture the complex patterns of user social relationships and personal behavior, and optimize access permission allocation. The study aims to enhance the model's understanding of the relationships between different nodes by integrating MLA mechanisms into GNN. Each layer of the attention mechanism enables the model to focus on different node characteristics, thereby enabling the model to more finely distinguish the importance of users and their associated objects, enhance the model's learning ability, improve its resolution of user preferences, and more accurately capture the user's true intentions. Consequently, this enhances the effectiveness of access control. The model structure is shown in Figure 4.

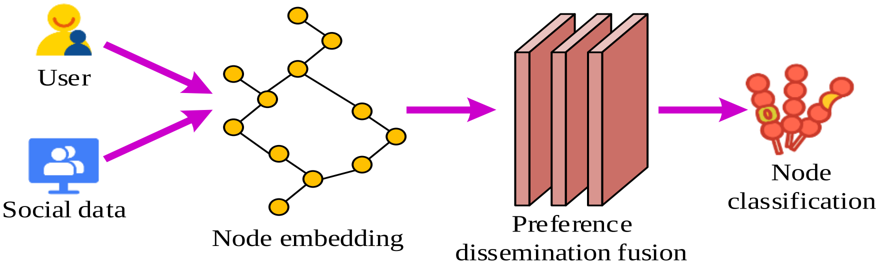


Figure 4: Access control method based on improved GNN.

In Figure 4, nodes are constructed based on users, their social friends, and social data posted by users. The user nodes include basic attributes, social relationship characteristics, behavior characteristics, and privacy setting characteristics. Social data nodes contain features of published content, interactive behavior, and social relationships. These features are transformed into low dimensional vectors through numerical processing and embedding learning, and embedded into GNNs as input vectors. Given the input data, the user's Social Preferences (SPs) and PPs are obtained, and the two preferences are fused. Then, the fused data is trained and the nodes are classified. The user attributes are selected to represent the user, and after embedding, the user node embedding matrix is obtained. The calculation method is shown in equation (6).

$$u_a = f(W_1 \times [P_a, E_a]) \quad (6)$$

In equation (6), u_a is the user node, P_a is the node embedding matrix, E_a is the node free embedding matrix, and W_1 is the node embedding weight. By using natural language processing tools to process and extract each social data, the embedding matrix of user posted social data nodes is obtained after embedding, and the calculation method is shown in equation (7).

$$d_i = f(W_2 \times [Q_i, V_i]) \quad (7)$$

In equation (7), d_i is a social data node. Q_i is the social data embedding matrix. V_i is the free embedding matrix of data nodes. W_2 is the embedding weight of social data. The embedded user nodes and social data nodes are input into the fusion layer and updated simultaneously through the MLA mechanism. The MLA mechanism calculation method is shown in equation (8).

$$A(Q,K,V) = \mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V \quad (8)$$

In equation (8), $A(Q,K,V)$ represents attention. Q, K, and V represent query, key, and value, respectively. T represents transpose. d_k represents the dimension of the key vector, used to scale dot product results and prevent gradient vanishing. The method for updating user SP nodes is shown in equation (9).

$$\begin{cases} p_a^{n+1} = u_a^n + \sum_{b \in S_a} \alpha_a^n u_b^n \\ \alpha_a^n = \mathrm{softmax}(\mathrm{Relu}(u_a^n)^T W_3 \mathrm{Relu}(u_b^n)) \end{cases} \quad (9)$$

In equation (9), p_a^{n+1} is the updated temporary node of user SPs. α_a^n is the attention score, which is the aggregated weight ratio of each neighboring node during node update. u_a^n and u_b^n are the n-th embeddings of nodes a and b. b represents all the explicit and implicit neighbor nodes of the node in the figure. $\mathrm{softmax}()$ and $\mathrm{Relu}()$ both represent activation functions. W_3 is the attention weight. The update of user PP nodes is shown in equation (10).

$$\begin{cases} q_a^{n+1} = u_a^n + \sum_{i \in C_a} \beta_a^n d_i^n \\ \beta_i^n = \mathrm{softmax}(MLP[u_a^n, d_i^n]) \end{cases} \quad (10)$$

In equation (10), q_a^{n+1} is the updated PP temporary node. β_a^n is the weight ratio of adjacent nodes when a user node updates. c_a is all user related data in the figure. MLP is a multi-layer perceptron. A multi-layer perceptron is a simple neural network used to perform nonlinear transformations on the feature vectors of nodes. It typically consists of multiple fully connected layers, each of which can be followed by a nonlinear activation function. The embedding of user nodes and social data nodes have different meanings in each dimension. If attention scores are calculated using functions such as dot product or mean pooling, it will result in inaccurate attention scores. Therefore, attention neural networks are used to calculate the attention scores of each neighboring node, and the results obtained by each neural network are finally normalized. The updated user's social preference temporary node and personal preference temporary node are weighted and fused to obtain the updated user node. The calculation method is shown in equation (11).

$$\begin{cases} u_a^{n+1} = \mu^{n+1} p_a^{n+1} + \delta^{n+1} q_a^{n+1} \\ \mu^{n+1} + \delta^{n+1} = 1 \end{cases} \quad (11)$$

In equation (11), u_a^{n+1} is the updated user node. μ^{n+1} and δ^{n+1} are the weights of SP temporary nodes and PP temporary nodes in the updated user nodes. Figure 5 shows the user preference fusion process.

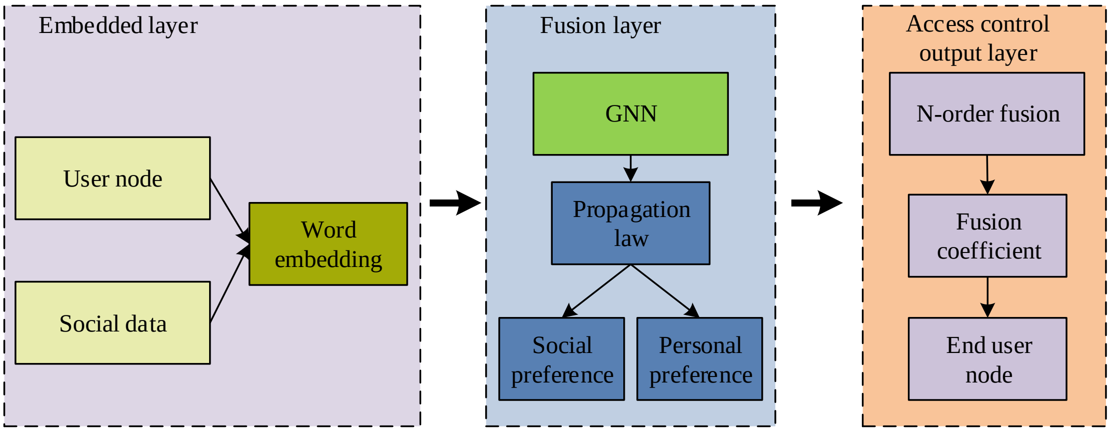


Figure 5: User preference fusion process.

In Figure 5, user preference fusion consists of three parts: embedding layer, preference propagation fusion layer, and access control output layer. In the embedding layer, word embeddings are performed on user nodes and social data as inputs to the model. In the preference propagation fusion layer, two GNNs simulate the propagation and change patterns of user social preferences among users and the propagation and change patterns of user personal preferences in social data. After N rounds of propagation and fusion, user nodes are updated using attention mechanisms based on explicit neighbor nodes, implicit neighbor nodes, and social data nodes. In the access control output layer, in the graph, N user nodes obtained through N preference propagation fusion are used to calculate the fusion coefficient through a linear neural network. Based on the fusion coefficient, the N user nodes are finally fused to obtain the final user nodes with user preferences. The fusion coefficient can quantify the importance or weight of user social preference temporary nodes and user personal preference temporary nodes. Therefore, the calculation of the fusion coefficient is completed by updating the user nodes and normalizing through a nonlinear transformation and a softmax function. At the same time, the user embedding vectors after each propagation are multiplied by their corresponding fusion coefficients, and these weighted embedding vectors are added to obtain the final user node vector. The calculation method is shown in equation (12).

$$\begin{cases} C_{u_a^n} = \mathrm{Softmax}(\tanh(W_4 u_a^n + e) \cdot d^T) \\ u_a = \mathrm{sigmod}(\sum_{n=1}^{N} C_{u_a^n} u_a^n) \end{cases} \quad (12)$$

In equation (12), $C_{u_a^n}$ is the fusion coefficient. W_4 is the fusion weight of user nodes. e is a natural constant. d is the dimension. $\mathrm{sigmod}()$ is the activation function. $\tanh()$ is a hyperbolic tangent function. Finally, the loss function of the model is defined to measure the difference between the predicted results of the model and the true labels, as expressed in equation (13).

$$L = \frac{1}{M} \sum_a -[y_a \cdot log(pr_a) + (1 - y_a) \cdot log(1 - pr_a)] \quad (13)$$

In equation (13), L is the loss and M is the amount of user nodes. y_a means the true label of the a -th user node, with a value of 0 or 1. When access is allowed, it is 1, and when access is prohibited, it is 0. pr_a is the probability of being judged as allowed access. During the training phase, the model will continuously adjust parameters based on the difference between the true labels and the predicted probabilities to minimize the loss function and optimize the probability estimates allowed for access. According to the loss function, all user nodes are classified based on whether they are allowed access, thus completing access control. The implementation process of the designed ACM is shown in Figure 6.

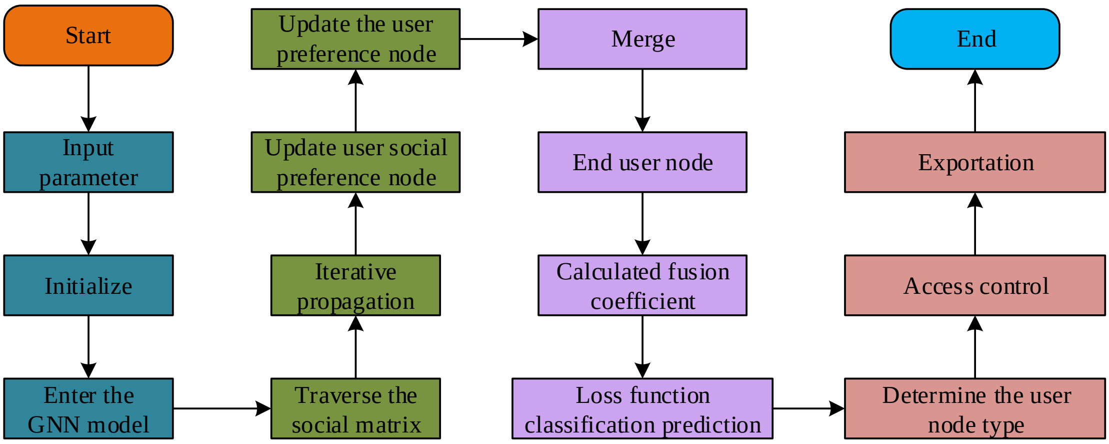

Start
Input parameter
Initialize
Enter the GNN model
Traverse the social matrix
Iterative propagation
Update user social preference node
Update the user preference node
Merge
End user node
Calculated fusion coefficient
Loss function classification prediction
Determine the user node type
Access control
Exportation
End

Figure 6: Implementation process of the proposed access control algorithm.

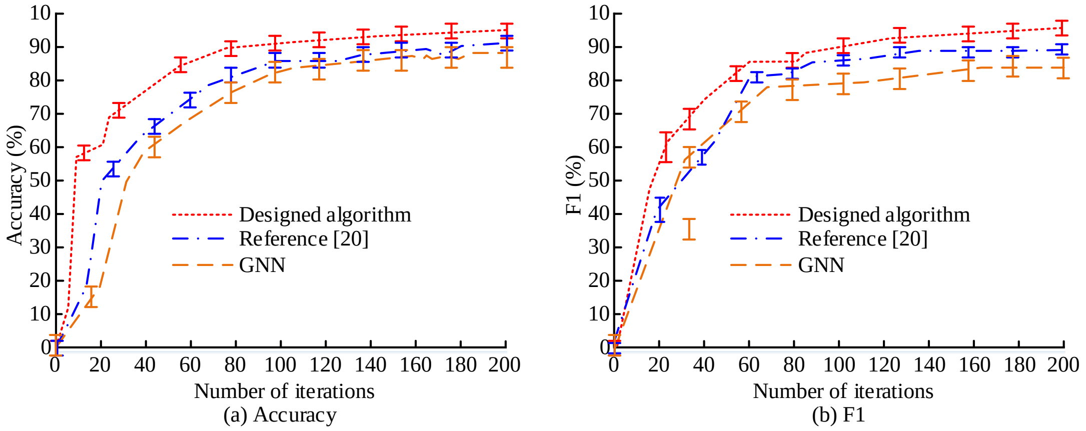
In Figure 6, the original data such as user information, social relationships, and content posted by users are first collected from social networks. Redundant information in the data is eliminated by removing duplicate items, and missing values are filled in to ensure the integrity of the data. The data are then uniformly converted through format conversion to complete the preprocessing of the collected data. The random seed is set to 42 using the random module and numpy library in Python, thereby ensuring that the generated random number sequence is the same every time the code is run. Then, the basic attributes and social behavior of users are extracted to construct user feature vectors, content feature vectors, and social relationship features. The model is used to map user node content to the status space, outputting user node embedding matrices and content node embedding matrices. Then, by analyzing users' social behavior, personal and social preferences are obtained, and a MLA mechanism is introduced to update the representations of user nodes and content nodes, highlighting the influence of important neighbors. Users' social and personal preferences are combined to form a comprehensive user preference representation. Next, using the fused user nodes as input, iterative propagation is performed through GNN to update the node representation. Next, a loss function is defined to measure the difference between the model's predicted results and the true labels, and the model parameters are adjusted to minimize the loss function. Finally, the trained model is employed to classify and predict new user nodes, determine whether to allow access to specific resources, and implement access control policies based on the results of access control decisions. The purpose of these actions is to ensure user privacy protection in social networks.

4 Analysis of ACM results on social networks

This chapter mainly elaborates on the experimental results of the UP-GNN-SNAC model. The first section is a performance analysis of ACMs based on improved GNN. The second section is an analysis of the practical application effect of the ACM based on improved GNN.

4.1 Performance testing of social network ACM

To verify the performance of the proposed UP-GNN-SNAC model, this study conducts simulation experiments using Python 3.7 on a Windows 11 64 bit operating system equipped with an Intel Core i7-14700KF central processor, 16GB of RAM, and 256GB of hard drive. The preferred propagation depth is 5, the learning rate is 0.0001, and the maximum number of iterations is 200. Accuracy is the most intuitive evaluation metric in classification models, representing the proportion of correctly classified samples to the total sample size. It measures the accuracy of the model in classifying user access permissions. The F1 value is the harmonic mean of accuracy and recall, used to comprehensively measure the performance of a model. It can balance the accuracy and recall of the model and avoid bias caused by imbalanced data. Firstly, the Twitter dataset is introduced to calculate the accuracy and F1 value of the research model, and compared with the accuracy and F1 value of traditional GNN and blockchain-based IoT ACMs in reference [20]. The results are shown in Figure 7.

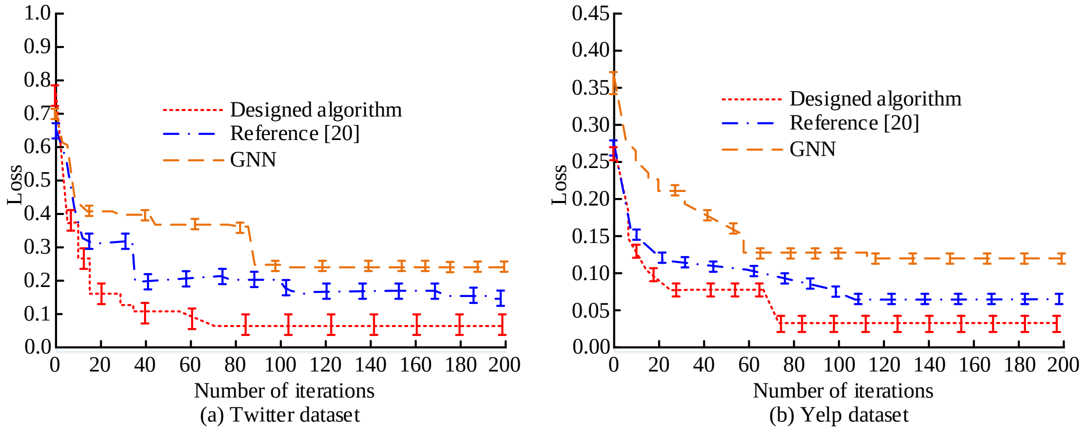


(a) Accuracy

(b) F1

Figure 7: Accuracy and F1 value of different models.

In Figure 7 (a), as the iteration increases, the accuracy of three models shows an upward trend. When iterating 200 times, the accuracy of traditional GNN is 88.3±1.97%, the accuracy of the model in reference [20] is 91.4±2.03%, and the accuracy of the research model is 95.7±2.11%. Compared with traditional GNN and the model in reference [20], the accuracy of the proposed ACM has been improved by 7.4% and 4.3%, respectively. In Figure 7 (b), as iterations increase, the F1 values of various models gradually increase and tend to flatten out. When the iteration reaches its maximum, the F1 values of GNN, the model in reference [20], and the research model are 83.6±1.16%, 89.8±1.09%, and 96.2±1.22%, respectively. Compared with the traditional GCN model and the model in reference [20], the F1 value of the proposed model has increased by 12.6% and 6.4%, respectively. The accuracy and F1 value of the research model are significantly higher, proving its high classification accuracy and good effectiveness. The loss function can be used to measure the difference between the model's predicted results and the true labels. The experiment then introduces the Yelp dataset and calculates the loss of the research algorithm under the Twitter and Yelp datasets. The results compared with the other two algorithms are shown in Figure 8.



(a) Twitter dataset

(b) Yelp dataset

Figure 8: Loss of different algorithms in different datasets.

In Figure 8 (a), on Twitter, as iterations increase, the losses of different models all show a decreasing trend. The loss values of traditional GNN, reference [20] models, and research models are 0.23±0.03, 0.14±0.02, and 0.07±0.03. In Figure 8 (b), the changes in loss curves of different models in Yelp are consistent with those in Twitter. The loss values of the three models are 0.12±0.02, 0.07±0.01, and 0.04±0.01. The loss value of the research model is greatly lower than others, indicating its good generalization ability. It performs well in different datasets, indicating good scalability. To further validate the performance of the proposed model by calculating its accuracy, recall, and Area Under the Curve (AUC) on both the Twitter and Reddit datasets, it is compared with traditional GNN, Graph Convolutional Network, signature schemes based on ciphertext policy attributes in reference [19], and models in reference [20]. The AUC metric is a statistical technique that can comprehensively reflect the model's ability to distinguish between different categories. A higher AUC value indicates that the model can more accurately predict which users should be granted access permissions, thereby reducing the likelihood of erroneously denying legitimate access or erroneously approving illegal access.

Meanwhile, the analysis of variance is used to evaluate the differences between models. ANOVA is a statistical method used to compare whether there is a significant difference in the mean between two or more groups. It is a widely used tool for researching experimental design and data analysis. ANOVA compares the variability between different groups to determine whether the within group variability is significantly smaller than the between group variability. If the inter-group variability is significantly greater than the intra-group variability, it can be concluded that there are significant differences between different groups. The significance level is set to 0.05. If $P<0.05$, it indicates that the difference between groups is statistically significant. Otherwise, the difference between groups is not statistically significant. The results are shown in Table 2.

Table 2: Precision, recall, and AUC values of different models.

Data set	Model	Precision	*P*	Recall	*P*	AUC	*P*
Twitter	GNN	0.782	<0.05	0.825	<0.05	0.791	<0.05
	Graph Convolutional Network	0.825		0.831		0.796	
	Reference [19]	0.865		0.904		0.836	
	Reference [20]	0.903		0.932		0.919	
	Designed algorithm	0.966		0.943		0.982	
Reddit	GNN	0.768	<0.05	0.813	<0.05	0.778	<0.05
	Graph Convolutional Network	0.821		0.846		0.815	
	Reference [19]	0.871		0.913		0.857	
	Reference [20]	0.896		0.935		0.911	
	Designed algorithm	0.972		0.938		0.976	

From Table 2, in the Twitter dataset, the accuracy, recall, and AUC values of the traditional GNN model are 0.782, 0.825, and 0.791, respectively. The three indicators of the graph convolutional network are 0.825, 0.831, and 0.796, respectively. The three indicators of the model in reference [19] are 0.865, 0.904, and 0.836, respectively. The three indicators of the model in reference [20] are 0.903, 0.932, and 0.919, respectively. The three indicators of the proposed model are 0.966, 0.943, and 0.982, respectively. On the Reddit dataset, the accuracies of the five models are 0.768, 0.821, 0.871, 0.896, and 0.972, respectively, with recall rates of 0.813, 0.846, 0.913, 0.935, and 0.938, and AUC values of 0.778, 0.815, 0.857, 0.911, and 0.976, respectively. In different datasets, the accuracy, recall, and AUC values of the three indicators of the proposed model are significantly higher than those of other models, and the differences between the three indicators of the five models are statistically significant ($P<0.05$), proving its good comprehensive performance and reliability. Finally, ablation experiments are conducted on the proposed model to calculate the accuracy, recall, F1 value, and running time of different modules. The results are shown in Table 3.

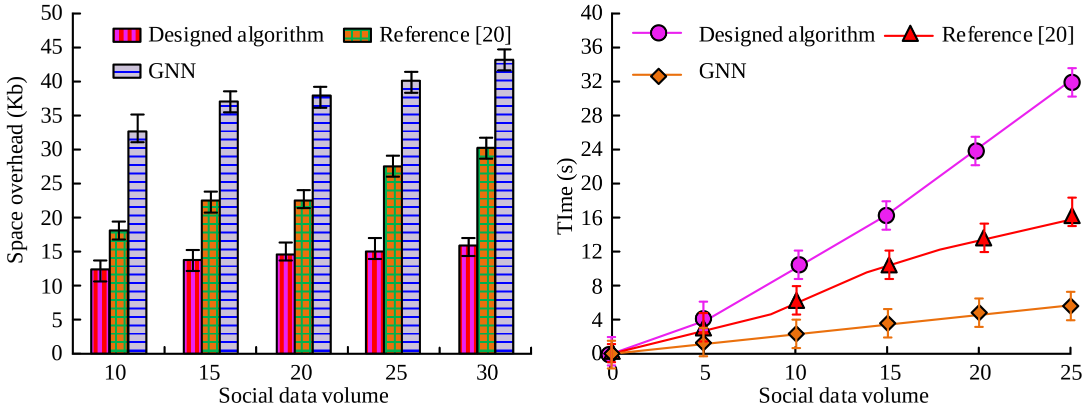
Table 3: Results of ablation experiment.

Module	Accuracy	Recall	F1	Running time (s)
Attention module	0.774	0.819	0.807	69.58
GNN module	0.862	0.842	0.853	43.12
Designed algorithm	0.973	0.956	0.961	46.89

From Table 3, the accuracy, recall, F1 value, and running time of the attention module are 0.774, 0.819, 0.807, and 69.58s, respectively. The accuracy, recall, F1 value, and running time of the GNN module are 0.862, 0.842, 0.853, and 43.12s, respectively. The four indicators of the designed model are 0.973, 0.956, 0.961, and 46.89s, respectively. The accuracy, recall, and F1 score of the designed model are higher than those of the two sub-modules, and the running time is higher than that of the attention module, but slightly lower than that of the GNN module. Despite the augmented computational complexity of the model, it has been demonstrated to enhance prediction accuracy. In practical application scenarios, the additional temporal expenditure is deemed justifiable.

4.2 Analysis of the practical application effect of ACM in social networks

To verify the practical application effect of the ACM based on improved GNN, this study first calculates the space overhead and computation time of the research model during encryption and decryption. It is compared with the results of traditional GNN and the model in reference [20]. Space overhead refers to the storage space occupied during the storage and operation of the model, as shown in Figure 9.



(a) Calculate expenses

(b) computing time

Figure 9: The computational cost and time of different models.

In Figure 9, as social data increases, the spatial overhead and computation time of different models gradually increase. When the social data scale is 30, the space overhead of traditional GNN, models in reference [20], and research models is 43.7±3.13Kb, 30.4±3.05Kb, and 16.2±2.88Kb, respectively, with computation times of 32.1±2.79s, 15.9±2.92s, and 5.3±0.97s. The space overhead and computation time of the research model are much lower than other models, which proves its high computational efficiency and low computational complexity. This study validates the access control effectiveness of the research model from seven aspects: User Preference Quantification (UPQ), Historical Records (HR), Privacy Metrics (PM), Sensitivity, User Attributes (UA), Trust, and Personalization. If the effect matches, output 1; otherwise, output 0. Table 4 compares the model with traditional GNN, reference [19], and [20]. Among them, UPQ can meet user needs, HRs are used to evaluate the consistency of user behavior, PMs and sensitivity can ensure data security and compliance, UAs can provide basic access control basis, trust can evaluate the reliability of user behavior, and personalization can improve user experience. The results are shown in Table 4.

Table 4: Access control effectiveness of different models.

Index	GNN	Reference [19]	Reference [20]	Research algorithm
UPQ	0	0	0	1
HR	1	1	1	1
PM	0	1	1	1
Sensitivity	1	1	0	1
UA	1	1	1	1
Trust level	1	1	0	1
Personalization	1	0	1	1

In Table 4, only the research model is consistent in terms of UPQ. In terms of HR and UA, all four models are consistent. In terms of PM, traditional GNN does not comply. The model in reference [20] does not match in terms of sensitivity and trustworthiness. This may be because the model may not have dynamically evaluated user behavior, authentication, or contextual information, resulting in an inability to accurately measure trust levels. In terms of personalization, only the model in reference [19] does not match. The research model is consistent in all 7 aspects, proving that its access control effect is relatively ideal. Finally, the Receiver Operating Characteristic curve (ROC) is introduced. The horizontal axis of the ROC curve represents the false positive rate, which represents the proportion of all negative samples that were incorrectly predicted as positive. The vertical axis represents the true sample rate, which represents the proportion of all actual positive samples correctly predicted as positive samples. The model correctly identifies requests that are actually positive samples as legitimate access and requests that are actually negative samples as illegal access. The ROC curves of the four models are calculated separately, and the results are shown in Figure 10.

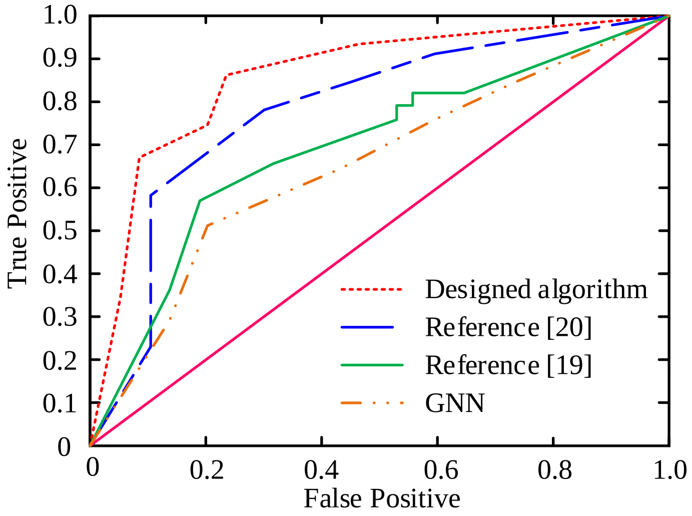


Figure 10: ROC curves and correlation coefficients R of four models.

From Figure 10, the ACM based on traditional GNN is closest to the standard line and has the smallest area under it. The lower area of the model in reference [19] is second, followed by the model in reference [20]. The ROC curve of the proposed model is closer to the upper left corner, with the largest lower area, indicating its strong classification ability and proving the high accuracy of the user preference-based ACM based on the improved GNN.

5 Discussion

The research aims to improve the effectiveness of social network access control by utilizing MLA mechanisms to enhance GNN's understanding of complex social relationships and personal behavior. A GNN-based social network ACM based on user preferences is proposed. The results showed that the accuracy and F1 value of the proposed ACM improved by 7.4% and 12.6% respectively compared to the GNN model and the blockchain-based IoT ACM, demonstrating its high classification accuracy. This is similar to the conclusion drawn by You M et al. [6], while the proposed model is superior. This is because the proposed model optimizes GNN through a MLA mechanism, which can more effectively capture complex patterns of user preferences and social relationships, thereby significantly improving performance. The computation time for encryption and decryption of the proposed model was 5.3 seconds, which was much lower than the GNN model and the blockchain-based IoT ACM. This conclusion is consistent with the findings of Gai K et al. [7], but the running efficiency of the proposed model is higher than that of the method proposed by Gai K et al. This is because the proposed model significantly improves computation time through MLA and information entropy. In summary, the proposed model performs well in multiple aspects. Although the proposed model can more accurately identify legitimate and illegitimate access through user preferences and privacy measurement mechanisms, effectively improving network security, it also increases certain computational overhead. Therefore, in practical applications, debugging needs to be carried out according to specific requirements.

6 Conclusion

ACM is crucial for the security of social networks, as it can help protect sensitive data and prevent malicious attacks and violations. To improve the accuracy and operational efficiency of social network ACM, a new type of ACM was designed based on the preferences of social network users. The user preferences were simulated using GNN, and a MLA mechanism was introduced to improve the model. The experimental results showed that the accuracy and F1 value of the proposed model were 95.7% and 96.2%, respectively, significantly higher than other models. This proved that through GNN and MLA mechanism, the model could dynamically capture user preference features and improve classification accuracy. The space cost of the proposed model was 16.2Kb and the computation time was 5.3s, which was significantly lower than the space cost and computation time of other models. This proved that the model adopted a lightweight GNN architecture, reducing computational complexity and optimizing algorithm design to reduce space cost. Although the proposed ACM has superior performance, there are still some shortcomings. The study did not test it on different types of social platforms, and future research will further test the performance of the model through different social network platforms to improve its universality. At the same time, the performance of the model in dynamic environments will be explored to cope with the constantly changing user behavior and data traffic in social networks.

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Competing interests

The authors have no relevant financial or non-financial interests to disclose.

Data availability statement

All data generated or analysed during this study are included in this article.

References

[1] Gai T, Cao M, Chiclana F, Zhang Z, Dong Y, Herrera-Viedma E, Wu J. (2023). Consensus-trust driven bidirectional feedback mechanism for improving consensus in social network large-group decision making. *Group Decision and Negotiation*, 32(1): 45-74. https://doi.org/10.1007/s10726-022-09798-7

[2] Kashmar N, Adda M, Ibrahim H. (2022). Access control metamodels: review, critical analysis, and research issues. *Journal of Ubiquitous Systems and Pervasive Networks*, 16(2): 93-102. https://doi.org/10.5383/JUSPN.03.01.000

[3] Wang W, Huang H, Yin Z, Gadekallu T R, Alazab, M, Su C. (2023). Smart contract token-based privacy-preserving access control system for industrial Internet of Things. *Digital Communications and Networks*, 2023, 9(2): 337-346. https://doi.org/10.1016/j.dcan.2022.10.005

[4] Thabit S, Yan L S, Tao Y, Abdullah A B. (2022). Trust management and data protection for online social networks. *IET Communications*, 2022, 16(12): 1355-1368. https://doi.org/10.1049/cmu2.12401

[5] Ameer S, Benson J, Sandhu R. (2022). Hybrid approaches (ABAC and RBAC) toward secure access control in smart home IoT. I*EEE Transactions on Dependable and Secure Computing*, 2022, 20(5): 4032-4051. https://doi.org/10.1109/TDSC.2022.3216297.

[6] You M, Yin J, Wang H, Cao J, Wang K, Miao Y, Bertino E. (2023). A knowledge graph empowered online learning framework for access control decision-making. *World Wide Web*, 2023, 26(2): 827-848. https://doi.org/10.1007/s11280-022-01076-5

[7] Gai K, She Y, Zhu L, Choo K K R, Wan Z. (2023). A blockchain-based access control scheme for zero trust cross-organizational data sharing. *ACM Transactions on Internet Technology*, 2023, 23(3): 1-25. https://doi.org/10.1145/3511899

[8] Wu H, Ye W, Guo Y. (2023). Data access control method of cloud network secure storage under Social Internet of Things environment. *International Journal of System Assurance Engineering and Management*, 2023, 14(4): 1379-1386. https://doi.org/10.1007/s13198-023-01942-z

[9] Azbeg K, Ouchetto O, Andaloussi S J. (2022). Access control and privacy-preserving blockchain-based system for diseases management. *IEEE Transactions on Computational Social Systems*, 2022, 10(4): 1515-1527. https://doi.org/10.1109/TCSS.2022.3186945

[10] Zhang L, Zhang Y, Wu Q, Mu Y, Rezaeibagha F. (2022). A secure and efficient decentralized access control scheme based on blockchain for vehicular social networks. *IEEE Internet of Things Journal*, 2022, 9(18): 17938-17952. https://doi.org/10.1109/JIOT.2022.3161047

[11] Zhao Y, Yu H, Liang Y, Conti M, Bazzi W, Ren Y. (2023). A sanitizable access control with policy-protection for vehicular social networks. *IEEE Transactions on Intelligent Transportation Systems*, 2023, 25(3): 2956-2965. https://doi.org/10.1109/TITS.2023.3285623

[12] Squicciarini A, Rajtmajer S, Gao Y, Semonsen J, Belmonte A, Agarwal P. (2022). An extended ultimatum game for multi-party access control in social networks. *ACM Transactions on the Web (TWEB)*, 2022, 16(3): 1-23. https://doi.org/10.1145/3555351

[13] Dixit M S, Wajgi M D, Wanjari S. (2022). Real time user access control on social network using deep learning. *International Journal for Research Publication and Seminar*, 2022, 13(2): 246-251. https://jrps.shodhsagar.com/index.php/j/article/view/598.

[14] Wen W, Fan J, Zhang Y, Fang Y. (2022). APCAS: Autonomous privacy control and authentication sharing in social networks. *IEEE Transactions on Computational Social Systems*, 2022, 10(6): 3169-3180. https://doi.org/10.1109/TCSS.2022.3218883

[15] Safi S M, Movaghar A, Ghorbani M. (2022). Privacy protection scheme for mobile social network. *Journal of King Saud University-Computer and Information Sciences*, 2022, 34(7): 4062-4074. https://doi.org/10.1016/j.jksuci.2022.05.011

[16] Ahmed F, Wei L, Niu Y, Zhao T, Zhang W, Zhang D, Dong W. (2022). Toward fine-grained access control and privacy protection for video sharing in media convergence environment. *International Journal of Intelligent Systems*, 2022, 37(5): 3025-3049. https://doi.org/10.1002/int.22810

[17] Salem R B, Aimeur E, Hage H. (2023). A multi-party agent for privacy preference elicitation. *Artificial Intelligence and Applications*, 2023, 1(2): 98-105. https://doi.org/10.47852/bonviewAIA2202514

[18] Mayeke N R, Arigbabu A T, Olaniyi O O, Okunleye O J, Adigwe C S. (2024). Evolving access control paradigms: A comprehensive multi-dimensional analysis of security risks and system assurance in

cyber engineering. *Asian Journal of Research in Computer Science*, 2024, 17(5): 108-124. https://doi.org/10.2139/ssrn.4752902

[19] Patil R Y. (2024). A secure privacy preserving and access control scheme for medical internet of things (MIoT) using attribute-based signcryption. *International Journal of Information Technology*, 2024, 16(1): 181-191. https://doi.org/10.1007/s41870-023-01569-0

[20] Zhonghua C, Goyal S B, Rajawat A S. (2024). Smart contracts attribute-based access control model for security & privacy of IoT system using blockchain and edge computing. *The Journal of Supercomputing*, 2024, 80(2): 1396-1425. https://doi.org/10.1007/s11227-023-05517-4