Design and Research of Accounting Automation Management System Based on Swarm Intelligence Algorithm and Deep Learning

Dan Gui[1], Wei Ma[2], Wanfei Chen[3]*
Financial Assets Department,
State Grid Henan Electric Power Company Information and Communication Branch, Zhengzhou 450000, China[1, 3]
Finance Department, State Grid Henan Electric Power Company, Zhengzhou 450000, China[2]

***Abstract*—In the current research, the application verification of traditional algorithms in actual accounting management is insufficient, and deep learning data processing capabilities need to be fully optimized in complex accounting scenarios. Given the challenges of efficiency and accuracy faced by the current accounting industry in the context of big data, this study creatively combines the swarm intelligence algorithm and deep learning technology to design and implement an efficient and accurate accounting automation management system. The research aims to investigate the potential of swarm intelligence algorithms and deep learning techniques in developing an automated accounting management system, with a focus on improving efficiency, accuracy, and scalability. Key research questions include exploring the optimal configuration of swarm intelligence algorithms for accounting tasks and assessing the performance of deep learning models in automating various accounting processes. Through experimental verification, the system is tested with the financial data of a large enterprise for three consecutive years. The results show that the system can significantly shorten the time of financial statement generation by 65%, reduce the error rate to less than 0.5%, and increase the accuracy of abnormal data recognition by as much as 90%. These data not only reflect the significant improvement of the efficiency and accuracy of the system but also prove its great potential in early warning of financial risk, providing intelligent and automated solutions for the accounting industry.**

***Keywords*—Swarm intelligence algorithm; deep learning; accounting; automation management**

I. Introduction

In today's wave of digital transformation, the accounting industry faces unprecedented challenges and opportunities [1, 2]. Traditional accounting processes, including data entry, account reconciliation, financial statement generation, etc., often rely on manual operations, which are time-consuming, labor-intensive, and prone to errors [3]. With the rapid development of big data, artificial intelligence, and other technologies, the emergence of accounting automation management systems provides new ideas and possibilities for solving these problems. Among them, the integration and application of swarm intelligence algorithms and deep learning technology are becoming a research hotspot in accounting automation management, opening up a new path for realizing the intelligence and automation of accounting work [4, 5].

Swarm intelligence algorithms, such as the ant colony algorithm and particle swarm optimization algorithm, are optimization algorithms that imitate the behavior of biological swarms in nature and have strong global search ability and robustness [6, 7]. In accounting automation management, swarm intelligence algorithms can effectively deal with complex decision-making problems, such as financial forecasting, cost control, etc., and find the optimal or approximately optimal solution by simulating the collaboration and competition of swarms [8]. Deep learning, as an essential branch of artificial intelligence, can automatically learn features from massive financial data through its powerful data processing and pattern recognition capabilities, realize automated account classification, anomaly detection, and other functions, and significantly improve the efficiency and accuracy of accounting work [9, 10].

However, there are still many challenges in applying swarm intelligence algorithms and deep learning technology to design accounting automation management systems [11]. How to design a reasonable algorithm model to adapt to the complexity and diversity of accounting data; How to ensure the stability and robustness of the algorithm and avoid decision-making errors caused by data fluctuations; How to realize the effective use of data on the premise of protecting data privacy is an urgent problem to be solved [12, 13]. In addition, developing and applying an accounting automation management system also involves compatibility with existing accounting software, user interface design, system security, and other issues, which require interdisciplinary knowledge and skills [14]. While there has been significant progress in the development of accounting automation systems, there are still several challenges and limitations that remain unaddressed. For instance, existing systems often lack the flexibility and scalability to handle complex accounting tasks and large datasets efficiently. Additionally, many systems rely on traditional rule-based approaches, which may not be well-suited for the dynamic and unpredictable nature of accounting data. In this paper, we propose a novel accounting automation management system based on swarm intelligence algorithms and deep learning techniques that aims to address these limitations and fill the existing gap in the field. Our system leverages the strengths of swarm intelligence for optimization and deep learning for pattern recognition, enabling it to handle complex accounting tasks with high accuracy and efficiency.

The purpose of this study is to deeply explore the key technologies and methods of accounting automation management system design based on swarm intelligence algorithm and deep learning and propose an efficient, intelligent, and safe accounting automation management solution through theoretical analysis and empirical research to provide theoretical guidance and practical reference for the digital transformation of the accounting industry. This research will focus on the following contents: First, analyze the application potential and limitations of swarm intelligence algorithm and deep learning in accounting automation management; The second is to design and implement the prototype of an accounting automation management system based on swarm intelligence algorithm and deep learning, and evaluate its performance and effect; The third is to put forward the algorithm optimization strategy according to the characteristics of accounting data to improve the intelligence level of the system; The fourth is to discuss the challenges and countermeasures of accounting automation management system in practical application, and provide direction for future research and practice.

The design and research of accounting automation management systems based on swarm intelligence algorithms and deep learning is not only an essential direction of technological innovation in the accounting field but also a key force in promoting the digital transformation of the accounting industry. Through this study, we expect to provide new ideas and methods for developing and applying accounting automation management systems, promote the intelligence and automation process of the accounting industry, and provide more powerful and intelligent tools for enterprise financial management and decision support. In the following part of this article, we will delve into the design and implementation of an accounting automation management system based on bee colony intelligence algorithms and deep learning technology. In Section II, we will first introduce the theoretical background and the research related to the research on accounting automation management strategy based on swarm intelligence algorithm, and then describe in detail the system architecture and method we propose in Section III. In Section IV, we will present the results of the experimental evaluation, demonstrating the effectiveness and efficiency of our system. Finally, in Section V, we will discuss the significance of our findings and suggest directions for future research. The paper is concluded in Section VI.

II. Research on Accounting Automation Management Strategy Based on Swarm Intelligence Algorithm

A. Ant Colony Algorithm

Social insects in nature, such as ants, show strong adaptability and flexibility and cooperate to complete tasks such as foraging and nesting by releasing up to 20 kinds of pheromones. These pheromones provide a means of navigation and communication for ants with limited vision, especially playing a key role in finding pathways back to the nest and dividing labor and cooperating [15]. Inspired by the social behavior of ants, scientists have developed ant colony algorithms, which provide new ideas for solving complex problems by simulating the action mechanism of pheromones. Years of studies have shown that ants can secrete a substance that affects the environment, promotes mutual communication or perceives environmental changes, namely pheromones. This discovery has deepened our understanding of ants' environmental interaction mechanisms [16, 17].

Imagine an ant facing two paths around obstacles. Because there is no former pheromone to guide it, it chooses randomly. During walking, ants release pheromones, which provide a selection basis for subsequent ants. Subsequently, ants are more inclined to choose the path with high pheromone concentration and supplement pheromone at the same time, and the pheromone concentration will naturally decay with time [18, 19]. Through the continuous selection and pheromone update of colony ants, pheromone space is formed to guide ants in making decisions.

The ant colony algorithm is based on an abstract model, which is shown in Fig. 1. Ants communicate and obtain environmental information through pheromones and perform tasks independently. The model assumptions include the following: environmental changes have feedback on ants; Pheromone is a medium for ants to communicate and obtain environmental information; Except for pheromones, behaviors are independent of each other and are not directly affected by other ants. The primary challenge of applying an ant colony algorithm is to transform the problem into an understandable form of an ant colony; that is, the problem needs to be properly described. After the description, an algorithm model based on the pheromone decision mechanism is developed. As the basis of ant communication and organization, the pheromone must simulate the use of the pheromone in the algorithm to guide ants' exploration path, just like real ants foraging and obstacle avoidance [20]. Renewal of pheromones directly controls ant colony behavior.

In applying the ant colony algorithm, heuristic information is defined as the basis of ant decision-making, which usually reflects prior knowledge. For example, in the traveling salesperson problem, the heuristic information is the reciprocal of the path distance. The longer the path, the smaller the information value, which affects the ant transition probability. Its calculation formula is shown in Eq. (1). In network problems, heuristic information correlates delay, bandwidth, and packet loss rate to guide the algorithm in finding the path that conforms to the network characteristics.

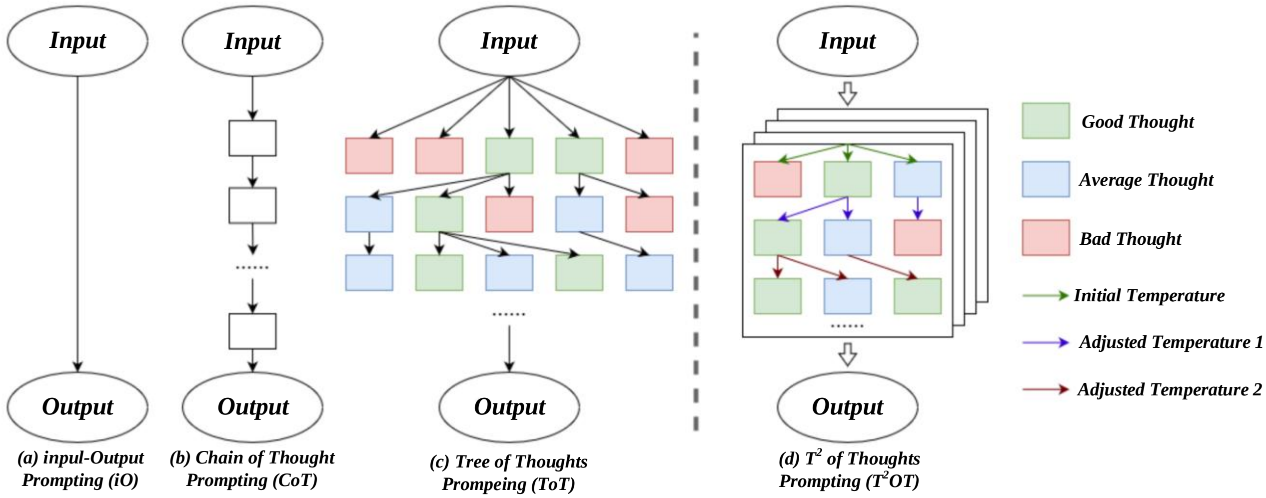

Fig. 1. Ant colony algorithm model.

$$p_{ij}^k = \begin{cases} \dfrac{\left[\tau_{ij}\right]^\alpha \left[\eta_{ij}\right]^\beta}{\sum\limits_{j \in allowed_k} \left[\tau_{ij}\right]^\alpha \left[\eta_{ij}\right]^\beta}, j \in allowed_k \\ 0,\text{otherwise} \end{cases} \tag{1}$$

In Eq. (1), ηij represents the heuristic information of the link (i, j), τij represents the pheromone concentration of the link (i, j), and allowedk represents the set of nodes that ant k can transfer to in the next step, a and β have the influence coefficients representing the pheromone concentration τij and the heuristic information ηij of network link (i, j), respectively. At the initial time, the pheromone concentration between each path is the same, and ηij (t) of the molecule is the heuristic information from node i to node j. Its expression is ηij (t) = 1/dij (t), where dij is the Euclidean distance between node i and node j, and the expression is Eq. (2):

$$d_{ij} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \tag{2}$$

(xi, yi) is a heuristic factor representing the relative importance of pheromone concentration; (xj, yj) is the visibility heuristic factor. The ant determines the next position by calculating the path transition probability and applying the wheel gambling method. After completing a round of searching, the path length is evaluated, and the shortest path is identified. Subsequently, the pheromone is updated according to the principle of "volatilization-release," as shown in Eq. (3) and Eq. (4).

$$\tau_{ij}\left(t+1\right) = \left(1-\rho\right) * \tau_{ij}\left(t\right) + \Delta\tau_{ij}^k \tag{3}$$

$$\tau^k = \sum_{k=1}^{m} \Delta\tau_{ij}^k \tag{4}$$

Where ρ is the pheromone volatilization factor, and the value range is ρ ∈ [0, 1]; m represents the number of ant colonies. Equation (5) represents the amount of pheromone left by all ants in the current search process, where the pheromone left by each ant in the current iteration process can be expressed as:

$$\Delta\tau_{ij}^k = \begin{cases} Q / L_k \quad Ki \ to \ Kj \\ 0, \ other \end{cases} \tag{5}$$

Among them, Q is a constant, which represents the total amount of pheromone that can be released in one search process of ant pheromone; Lk represents the total length of the path traveled by ant k in this cycle, and Ki to Kj represents a range sequence.

B. Drosophila Algorithm

Set the population size (popsize) and the maximum number of iterations (maxgen). Among them, (Xaxis; Yaxis) represents the two-dimensional coordinates of each individual in the fruit fly community, LR represents the position range of the fruit fly population, and the mathematical expression of the initial position is as follows (6)-(7):

$$X_{axis} = rand(LR) \tag{6}$$

$$Y_{axis} = rand(LR) \tag{7}$$

Rand is a function used to generate random numbers. When individuals in the community search for food, their flight direction and distance are random. The following expression represents the new position when the fruit fly i flies to the next moment, as shown in Eq. (8)-(9):

$$X_i = X_{axis} + rand(FR) \tag{8}$$

$$Y_i = Y_{axis} + rand(FR) \tag{9}$$

FR denotes the range of a single flight, and axis refers to the straight line in the coordinate system. Because the specific location of the food source is unknown, first use the following formula to calculate the distance Disti of the individual fruit fly from the origin:

$$Dist_i = \sqrt{X_i^2 + Y_i^2} \tag{10}$$

Then, the taste concentration determination value Si is calculated by the following Eq. (11):

$$S_i = 1 / Dist_i \tag{11}$$

Fitness refers to the quality or quality measure of a solution. The taste concentration value Smelli of each individual in the current population is expressed by the following Eq. (12):

$$Smell_i = fitness(S_i) \tag{12}$$

C. Ant Colony Algorithm for Path Preprocessing of Drosophila Algorithm

When the traditional ant colony algorithm searches the path, the node traversal probability is equal, leading to a blind search in the initial stage [21]. In order to solve this problem, the Drosophila algorithm is introduced to pre-plan the path, which is transformed into the pheromone required by the ant colony algorithm, and the ant colony algorithm is guided to avoid blind search, reduce node traversal, and shorten the running time [22]. After pre-planning, the ant colony algorithm optimizes the search on this basis.

According to the requirements, FOA (Fruit Fly Optimization Algorithm) is first used for path preprocessing, and then the preprocessed path is converted into a pheromone, which is imported into the ant colony algorithm (ACO). The calculation formula is shown in Eq. (13).

$$\tau(i,j) = \tau^s(i,j) + \Delta\tau^B(i,j) \tag{13}$$

Where τ (i, j) represents the pheromone concentration from node i to node j, τs (i, j) is the original pheromone from node i to node j, and τB (i, j) refers to the pheromone increment from node i to node j converting the results of FOA search.

The ant uses the roulette method and selects the next node according to the formula. After the algorithm is finished, the pheromone is updated, and a portion is added and volatilized. After each generation of ants iterates, the paths are compared, and the shortest path is determined when the set number of iterations is reached.

III. Design of Accounting Automation Management System Based on Swarm Intelligence Algorithm and Deep Learning

A. Deep Learning Neural Network Technology

In the study, the performance and effectiveness of the accounting automation management system were evaluated in depth, and the comprehensive functionality, data accuracy and operational efficiency of the accounting automation management system were compared with popular accounting and financial software solutions in the market (such as QuickBooks, Xero, SAP, Oracle Financials and Kingdee, which may include other software) in order to fully verify the performance of the system in the real financial environment. To achieve this assessment, the accounting automation management system was systematically integrated with the above-mentioned software and extensive testing and data analysis were implemented. In terms of text processing of accounting sheets, special attention is paid to text records containing complex information such as equipment numbers, timestamps, exception reports, etc., and through preprocessing (including text segmentation, data cleaning, format standardization) and feature extraction (using bag-of-word model BoW, word frequency-inverse document frequency TF-IDF, N-gram model and word embedding technology, etc., the word embedding technology effectively solves the problems of dimensional disaster, data sparsity and semantic loss caused by high-dimensional data by mapping words or phrases to real low-dimensional vectors [25]), which successfully extracted critical information, which was critical for subsequent root cause analysis and ticket recommendations [23, 24]. In addition, the autoencoder (AE) deep learning technology is introduced to compress the high-dimensional input data into the low-dimensional feature vector space by using its powerful feature extraction ability [26], and the applications of denoising autoencoder (to enhance the robustness of the model), sparse autoencoder (to improve the compression ratio and learn the data structure), and variational autoencoder (to maximize the probability of data sample union, optimize the model parameters and hidden variables a posteriori, and endow the model generation ability and latent distribution simulation ability) are explored.

Neural network attention is introduced into the design of an accounting automation management system, which makes the network automatically focus on crucial information and improves its performance and interpretability. It is divided into three categories: soft attention (global), intricate attention (local), and self-attention (internal attention) [27, 28]. Soft attention calculates the weights based on similarity, utilizing all the input information, but it is computationally heavy. Intricate attention focuses on local areas, reduces computation, and conforms to visual characteristics but may ignore critical information. Self-attention considers inter-input and inter-output relationships and enhances long-distance dependency and structure capture. Principle of attention mechanism: When generating output, the model selectively focuses on relevant parts of the input according to the context, assigns different weights, optimizes information coding, and improves accuracy and robustness. The steps include calculating weights, such as dot product, additivity, self-attention, etc.; Apply weights to the weighted average or splice the inputs; Update the output, direct output, or further process.

In order to optimize the feature extraction capability, this study applies convolution operation to graph structure to realize graph data feature learning and classification. Unlike CNN in regular grid processing, GCN (Graph Convolution Networks) is good at unstructured graph data. The advantages are that it deals with complex graph structures and has good interpretability and visualization. The adjacency matrix expresses the graph structure, vectorizes node and edge features, and updates the features by convolution. Through the operation of the adjacency matrix and node feature matrix, feature aggregation and transfer are realized, and local features are extracted by convolution, similar to CNN. GCN also contains pooling operations for dimension reduction and feature abstraction of graph data.

B. Model Building

Automated accounting analysis aims to quickly and accurately identify and solve system failures and improve system stability and reliability. Analysis methods are divided into traditional and machine learning categories. Traditional methods rely on manual system performance analysis or use tools such as FMEA to identify anomalies. However, their ability to process large-scale data and quickly locate anomalies is limited and error-prone [29, 30]. The machine learning method builds a model by analyzing the system structure and historical data and automatically identifying anomalies' root causes. It has the advantages of processing massive data, quickly analyzing, and automatically adjusting the model to improve accuracy and real-time response, significantly superior to traditional manual methods.

The model is shown in Fig. 2. First, the preprocessed node data is vectorized for model training. The words in the node are split into characters, and a 70-dimensional hot encoding represents each character. The encoding table contains 26 letters, ten digits, 33 unique characters, and line breaks. Each node is transformed into a matrix of 70. Excessively long characters are truncated, and more characters should be filled with zero vectors. The encoding order starts from the last character, which is convenient for the weight association of fully connected layers. The vectorized node data is input into the model training. The model contains nine layers of neural network: 6 one-dimensional convolutional layers and three fully connected layers. A maximum pooling layer follows the first, second, and last convolutional layers. In the model, the kernel size of the first two convolution layers is 7, the core size of the last four layers is 3, each layer has 256 kernels, and the size of the pooling layer is 3. The node data outputs a 1008 × 256 feature map through the first convolutional layer. After the first pooling layer, the feature map size is 336 × 256.

Text feature extraction only uses work order data, ignoring the relationship between nodes. GCN model can handle graph structure, learn nodes and relationship features, and be used for root cause analysis. GCN training requires node features and an adjacency matrix. The feature extraction method is the same as before, and the top 200 high-frequency words are selected. The adjacency matrix represents the node relationship, and the weight is the number of occurrences of dependent paths in the historical work order description. GCN consists of two graph convolutional layers and a fully connected layer. Node features and adjacency matrix input convolutional layer, activated by ReLU, and fully connected layer and SoftMax output node features. The node features and graph features extracted from the text are used to match the abnormal root causes of the work order. Preprocess the exception description of the current work order, convert it into a character-level vectorized representation, and obtain the work order feature vector. By calculating the comprehensive similarity between the work order and the node, the node with the highest similarity is selected as the abnormal node. The root cause analysis is output by classification, and the similarity between the work order and each node is given, and the probability of abnormal nodes is calculated accordingly.

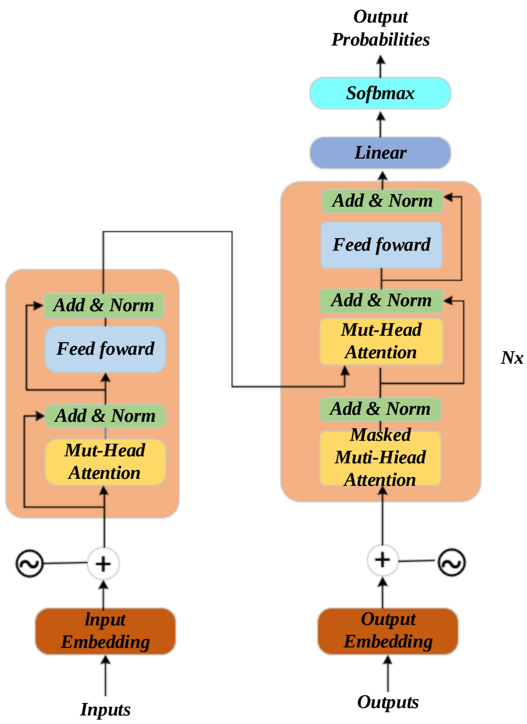

Fig. 2. Input data processing model.

IV. Experimental Results and Analysis

Fig. 3 compares the algorithm's performance at a confidence level of 90%. The model based on swarm intelligence algorithm and deep learning performs outstandingly in two-thirds of the dimension of ROC, thanks to its utilization of second-order difference information, which improves search efficiency. The algorithm also has advantages in the extreme dimensions of ES and entropy. The model continues to show competitiveness at higher confidence levels (95%, 97.5%, 99%).

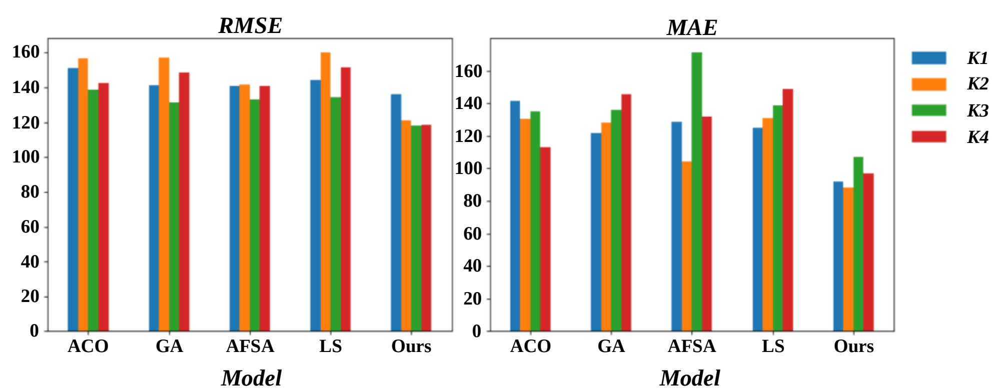

Fig. 3. Model results.

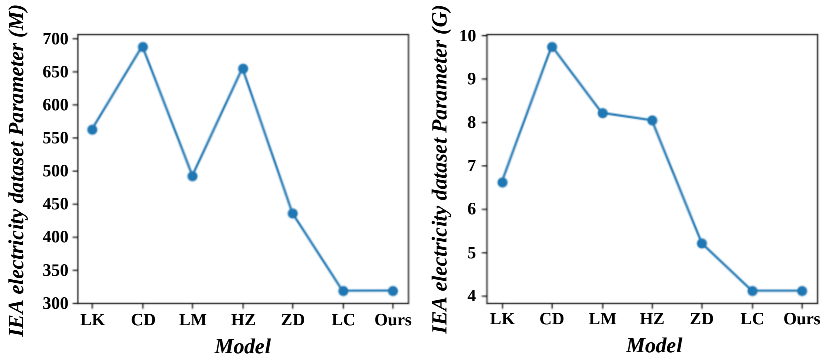

Fig. 4. Results at different confidence levels.

In Fig. 4, the HV value based on the swarm intelligence algorithm and deep learning model is the highest, and the second-order differential evolution operator effectively broadens the frontier of the Pareto solution set and enhances the extensibility of understanding. Table I shows that GBDT is the best among traditional methods but not as good as deep learning. CharCNN-based TicketRCA is better than CNN and RNN models. Due to the lack of a timing series of work order data keywords, models such as TextCNN are limited. CharCNN character-level representation is better. After the combination of GCN, TicketRCA root cause analysis is significantly improved, proving that the graph model is adequate. Although TicketMining and NetSieve are better than traditional machine learning, they are not as good as TicketRCA, indicating that deep learning can better extract text features.

In Fig. 5, Four strategies construct the anchor node table: the top 100 is selected by degree sorting, the top 100 by PageRank sorting (d = 0.9), and 100 is randomly selected. The joint strategy combines the top three in a ratio of 4: 4: 2 (degree sorting 40, PageRank 40, random 20). Experiments show that the accuracy of degree ranking and PageRank strategy is similar. The stochastic strategy also performs similarly, and the joint strategy has the highest accuracy, indicating that the combination of centrality and random noise is beneficial to improve the node classification performance.

In the ablation experiment, the joint strategy is used to construct the anchor node table and generate the subgraph sequence. Fig. 6 shows that the complete model samples ten anchor nodes and distances, three neighbors and relationships, one self-node, and a self-ring edge. The ablation model does not sample anchor nodes, neighbors, self-nodes, and self-ring edges. The complete model is better than the Transformer, which samples the whole graph. In ablation, unsampled anchor nodes have the most significant influence, followed by neighbors, and self-nodes and self-ring edges have less influence.

TABLE I. MODEL COMPARISON OF RESULTS

	Accuracy	Precision	Recall	F1-score
SVM	0.3311	0.2717	0.3311	0.2706
DecisionTree	0.3872	0.2519	0.3872	0.2904
RandomForest	0.3971	0.3014	0.3971	0.3179
GBDT	0.5214	0.4488	0.5126	0.4642

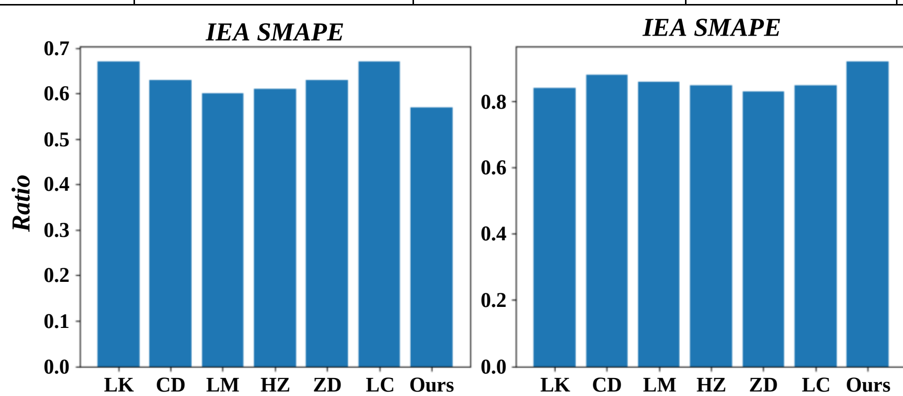

Fig. 5. Comparison of node classification accuracy of anchor node sampling methods based on different strategies.

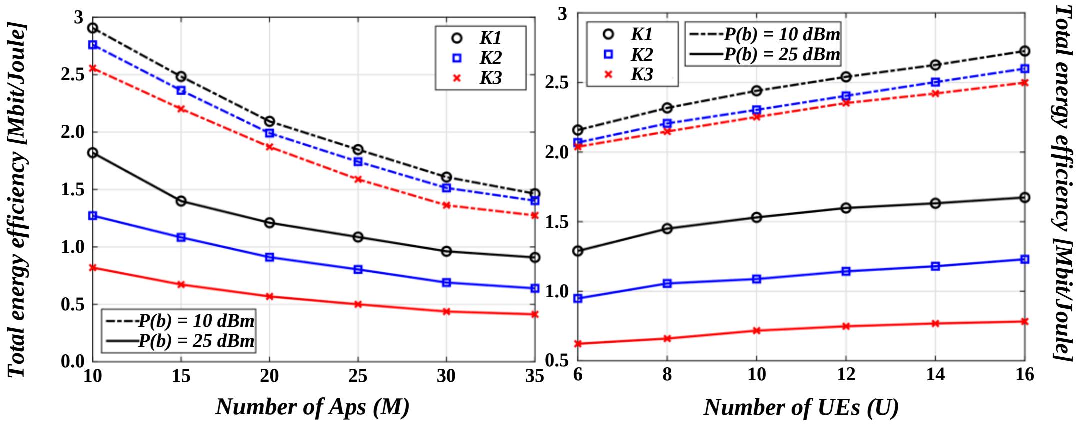

Fig. 6. Comparison of node classification accuracy based on different sampling objects.

The hyperparameter analysis is shown in Fig. 7. The data set performs best when the anchor node table size is 100 and 10 anchor nodes are sampled. Increasing the number of anchor nodes usually improves accuracy, but performance degrades after more than 10-20. The performance is optimal when the anchor node table size is 100. In order to improve system performance, more anchor nodes are needed to cover a wide range of knowledge fields due to the vast number of entities and relationships. When using large and complex data sets, it is necessary to build a larger anchor node table, sample more anchor nodes, and select appropriate strategies to ensure representativeness and improve model performance and robustness.

Fig. 8 shows the impact of the number of convolution kernels on predictive analysis. Using a data set with rich relationship types, its 237 relationships are denser. The results support the conjecture. more convolution kernels and high-dimensional relational embeddings improve performance. Since 128-dimensional embeddings cannot fully describe features, high-dimensional embeddings require broad convolution support.

Fig. 9 shows that on the sparse graph, the experimental results decrease by 5% and 9%, respectively, highlighting the improvement of the method in prediction accuracy. Removing both mechanisms significantly decreased accuracy. The embedding quality may only depend on the relationship embedding of small graphs near the target node. One-hop relationship is more important than neighbor nodes. Rich relationship types generate more combinations and provide nodes with more expressive subgraph features. Two-dimensional convolution and Transformer encoder fuse relationships and node embeddings. Therefore, compared with FB15k-237, which has 20 times more types of relationships, the performance degradation of WN18RR is more evident without these two modules.

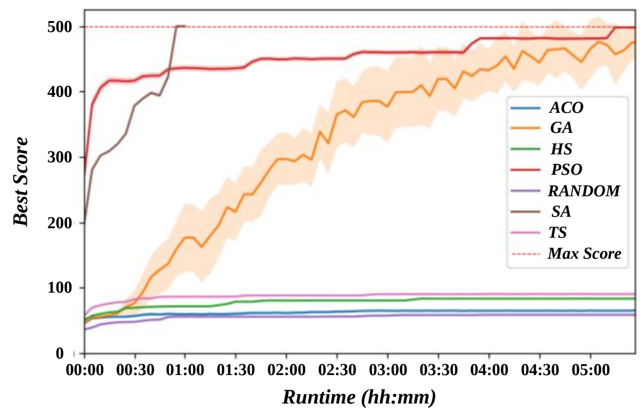

Fig. 7. Hyperparameter analysis.

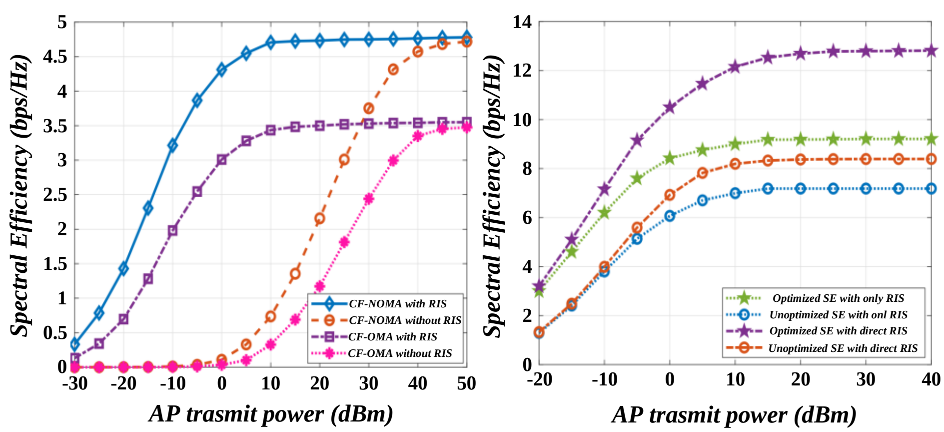

Fig. 8. Effect of the number of convolution kernels on predictive analysis.

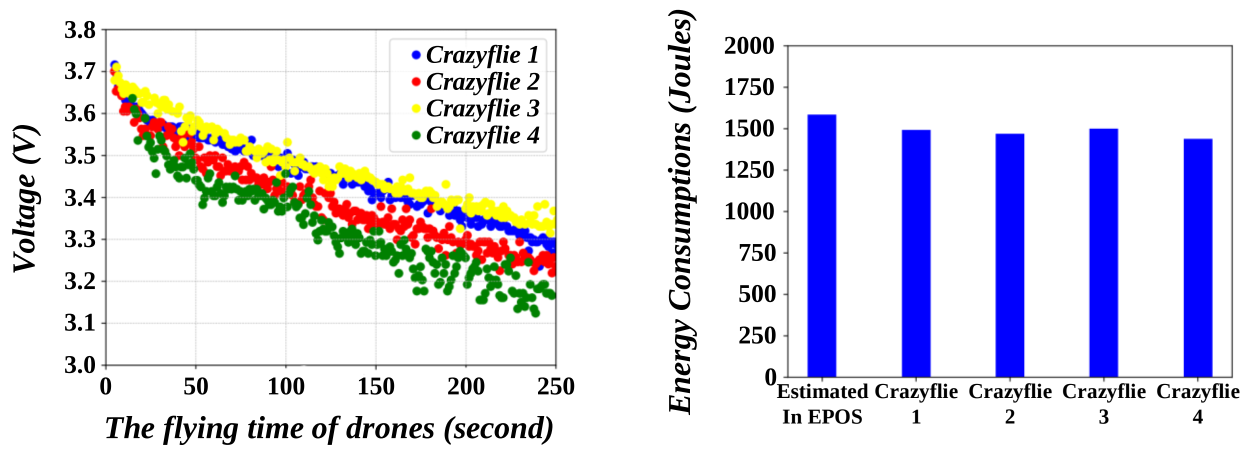

Fig. 9. Experimental results of sparse graph.

Fig. 10 shows that the AUC values of all models exceed 0.5, indicating that the prediction is better than a random guess and has data fitting ability for M&A prediction. The linear regression AUC value is high, but the F1 score is lower than that of decision tree regression, reflecting its foundation and low robustness. The logistic regression AUC reached 0.65, which was the best performance, thanks to the non-linear fit. The decision tree regression AUC was only 0.551, which was poorly fitted. AdaBoost boosts AUC but is limited by weak classifier performance. Through high-dimensional mapping and the kernel trick, the AUC of SVM is 0.014 higher than AdaBoost's. The overall effect of the machine learning method is not good, mainly due to insufficient capture of M&A data distribution information. Baseline performed the worst among the deep learning models, with an AUC of about 0.5. LSTM is significantly improved, AUC super logistic regression 0.04, good at long sequence data, capturing data connections. The model proposed in this paper has an AUC of 0.721, the best ACC and F1Score, which are 0.071 and 0.031 higher than logistic regression and LSTM, respectively, indicating that it can effectively fit the prediction data and achieve accurate prediction.

Fig. 11 shows the loss curve of the ablation experimental model, which intuitively reflects the prediction accuracy. The AUC value of AttDNN was 0.721, with the largest area of the ROC curve, followed closely by LSTM, with a difference of 3.1 percentage points in AUC values. Logistic regression AUC is higher and marked yellow. The AUC of linear regression, decision tree, AdaBoost, and SVM are low, the prediction effect is poor, and the ROC curves almost coincide. As the basic algorithm of deep learning, the Baseline lacks a regularization layer and attention mechanism training. Its ROC curve is close to the (FPR = 1, TPR = 1) connection of random guess, and the prediction effect is equivalent to random.

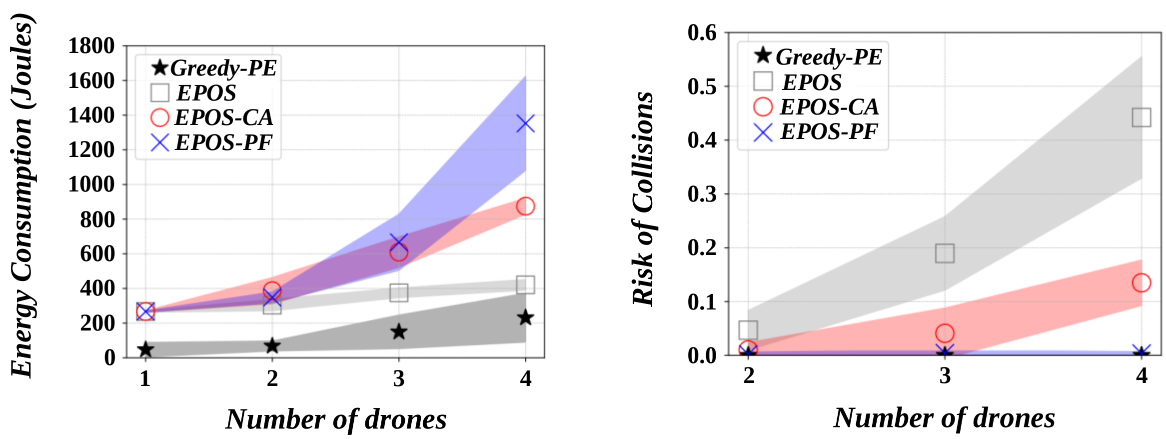

Fig. 10. Results of ablation experiment.

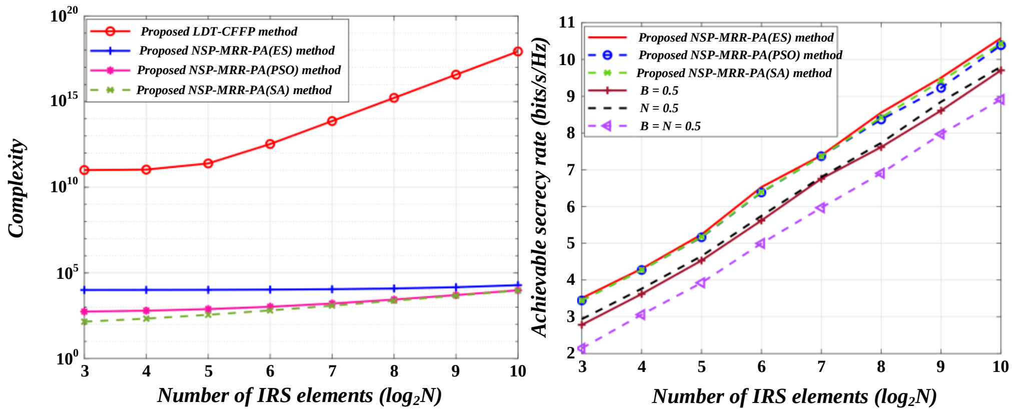

Fig. 11. Loss curve of ablation experimental model.

V. Discussion

Our results demonstrate the effectiveness and efficiency of our proposed system in handling complex accounting tasks and large datasets. However, there are still several challenges and limitations that need to be addressed in future work. For instance, while our system has shown promising performance in controlled environments, its robustness and scalability in real-world scenarios remain to be tested. Additionally, there is a need for further research on the integration of our system with existing accounting software and platforms to facilitate seamless adoption and use. In conclusion, our work represents a significant step forward in the field of accounting automation, and we believe that it provides a solid foundation for future research and development efforts.

VI. Conclusion

In the context of the digital transformation of the accounting industry, this study creatively combines a swarm intelligence algorithm with deep learning technology to design and develop an intelligent accounting automation management system that aims to address the limitations of traditional accounting management in big data processing.

By applying swarm intelligence algorithms and deep learning technology to accounting automation management, this study significantly improves the efficiency and accuracy of accounting work and enhances the ability to detect early warnings of financial risk. Experimental data show that the system's efficiency in processing financial statements has increased by 65%, mainly due to the optimization path of the swarm intelligence algorithm and the intelligent analysis ability of deep learning, which significantly shortens the data processing cycle.

This study provides an intelligent and automated solution for the accounting industry, and the accuracy rate is greatly improved. The error rate is reduced to less than 0.5%, indicating that the system can effectively identify and reduce human errors when processing complex financial data, significantly improving the accuracy of data processing.

Through continuous training of deep learning models, the system's recognition accuracy of abnormal data has reached 90%, providing strong technical support for financial risk early warning and effectively reducing potential financial risks.

This study designed an accounting automation management system based on swarm intelligence algorithms and deep learning, which is expected to improve accounting processes' efficiency and accuracy significantly. However, this study has limitations: the system was only tested in a controlled environment, and the experimental dataset cannot cover all accounting task scenarios. Subsequent research requires more extensive testing in real-world scenarios, using more diverse datasets and further exploring integration with existing accounting software platforms to develop complex swarm intelligence algorithms and deep learning models to enhance system performance.

References

[1] J. E. Gerken et al., "Geometric deep learning and equivariant neural networks," Artificial Intelligence Review, vol. 56, no. 12, pp. 14605-14662, 2023.

[2] M. S. Hashish, H. M. Hasanien, Z. Ullah, A. Alkuhayli, and A. O. Badr, "Giant Trevally Optimization Approach for Probabilistic Optimal Power Flow of Power Systems Including Renewable Energy Systems Uncertainty," Sustainability, vol. 15, no. 18, 2023.

[3] S. Ferilli, E. Bernasconi, D. Di Pierro, and D. Redavid, "A Graph DB-Based Solution for Semantic Technologies in the Future Internet," Future Internet, vol. 15, no. 10, 2023.

[4] C. Cavallaro, C. Crespi, V. Cutello, M. Pavone, and F. Zito, "Group Dynamics in Memory-Enhanced Ant Colonies: The Influence of Colony Division on a Maze Navigation Problem," Algorithms, vol. 17, no. 2, 2024.

[5] C. L. Galimberti, L. Furieri, L. Xu, and G. Ferrari-Trecate, "Hamiltonian Deep Neural Networks Guaranteeing Nonvanishing Gradients by Design," Ieee Transactions on Automatic Control, vol. 68, no. 5, pp. 3155-3162, 2023.

[6] J. Yu, X. You, and S. Liu, "A heterogeneous guided ant colony algorithm based on space explosion and long-short memory," Applied Soft Computing, vol. 113, 2021.

[7] F. Zhang, Z. Gao, J. Huang, P. Zhen, H.-B. Chen, and J. Yan, "HFOD: A hardware-friendly quantization method for object detection on embedded FPGAs," IEICE Electronics Express, vol. 19, no. 8, 2022.

[8] X. Wang, Z. Fu, and X. Li, "A Graph Deep Learning-Based Fault Detection and Positioning Method for Internet Communication Networks," IEEE Access, vol. 11, pp. 102261-102270, 2023.

[9] H. Chen and H. Eldardiry, "Graph Time-series Modeling in Deep Learning: A Survey," Acm Transactions on Knowledge Discovery from Data, vol. 18, no. 5, 2024.

[10] L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, "Hands-On Bayesian Neural Networks-A Tutorial for Deep Learning Users," IEEE Computational Intelligence Magazine, vol. 17, no. 2, pp. 29-48, 2022.

[11] H. Liu, B. Hu, and Y. Cao, "HDMA-CGAN: Advancing Image Style Transfer with Deep Learning," International Journal of Pattern Recognition and Artificial Intelligence, vol. 38, no. 09, 2024.

[12] J. Fu, Z. Yang, M. Liu, H. Zhang, and Y. Zhang, "Highly-efficient design method for coding metasurfaces based on deep learning," Optics Communications, vol. 529, 2023.

[13] Emilio Abad-Segura, Alfonso Infante-Moro, Mariana-Daniela González-Zamar, and Eloy López-Meneses, "Influential factors for a secure perception of accounting management with blockchain technology," Journal of Open Innovation: Technology, Market, and Complexity, vol. 10, no. 2, pp. 100264, 2024.

[14] Erik S. Boyle, "How do auditors' use of industry norms differentially impact management evaluations of audit quality under principles-based and rules-based accounting standards?" Journal of International Accounting, Auditing and Taxation, vol. 54, pp. 100598, 2024.

[15] Elsa Pedroso and Carlos F. Gomes, "Disentangling the effects of top management on management accounting systems utilization," International Journal of Accounting Information Systems, vol. 53, pp. 100678, 2024.

[16] Fangjuan Qiu, Nan Hu, Peng Liang, and Kevin Dow, "Measuring management accounting practices using textual analysis," Management Accounting Research, vol. 58, pp. 100818, 2023.

[17] Linda Hui Shi, Kristin Brandl, Jing Song, and Shaoming Zou, "Global account management: Knowledge resources and capabilities for relationship management," International Business Review, vol. 33, no. 5, pp. 102315, 2024.

[18] Haiyan Sun, "Construction of integration path of management accounting and financial accounting based on big data analysis," Optik, vol. 272, pp. 170321, 2023.

[19] Esra Gülmez, Halil Ibrahim Koruca, Mehmet Emin Aydin, and Kemal Burak Urganci, "Heuristic and swarm intelligence algorithms for work-life balance problem," Computers & Industrial Engineering, vol. 187, pp. 109857, 2024.

[20] Gang Hu, Feiyang Huang, Kang Chen, and Guo Wei, "MNEARO: A meta swarm intelligence optimization algorithm for engineering applications," Computer Methods in Applied Mechanics and Engineering, vol. 419, pp. 116664, 2024.

[21] Su Hu and Hua Yin, "Research on the optimum synchronous network search data extraction based on swarm intelligence algorithm," Future Generation Computer Systems, vol. 125, pp. 151-155, 2021.

[22] Lifu Ding, Youkai Cui, Gangfeng Yan, Yaojia Huang, and Zhen Fan, "Distributed energy management of multi-area integrated energy system based on multi-agent deep reinforcement learning," International Journal of Electrical Power & Energy Systems, vol. 157, pp. 109867, 2024.

[23] Lukáš Klein, Ivan Zelinka, and David Seidl, "Optimizing parameters in swarm intelligence using reinforcement learning: An application of Proximal Policy Optimization to the iSOMA algorithm," Swarm and Evolutionary Computation, vol. 85, pp. 101487, 2024.

[24] Li Sheng Kong, Muhammed Basheer Jasser, Samuel-Soma M. Ajibade, and Ali Wagdy Mohamed, "A systematic review on software reliability prediction via swarm intelligence algorithms," Journal of King Saud University - Computer and Information Sciences, vol. 36, no. 7, pp. 102132, 2024.

[25] Xianfang Liu, "Mathematical scheduling model of complex industrial process combining swarm intelligence algorithm and swarm dimension reduction technology," Results in Engineering, vol. 21, pp. 101796, 2024.

[26] Qirat Nizamani et al., "Nature-inspired swarm intelligence algorithms for optimal distributed generation allocation: A comprehensive review for minimizing power losses in distribution networks," Alexandria Engineering Journal, vol. 105, pp. 692-723, 2024.

[27] Haoxin Wang and Libao Shi, "A multi-direction guided mutation-driven stable swarm intelligence algorithm with translation and rotation invariance for global optimization," Applied Soft Computing, vol. 159, pp. 111614, 2024.

[28] Ahmad Alferidi, Mohammed Alsolami, Badr Lami, and Sami Ben Slama, "Design and implementation of an indoor environment management system using a deep reinforcement learning approach," Ain Shams Engineering Journal, vol. 14, no. 11, pp. 102534, 2023.

[29] Frank Bodendorf and Jörg Franke, "Synthesis of activity-based costing and deep learning to support cost management: A case study in the automotive industry," Computers & Industrial Engineering, vol. 196, pp. 110449, 2024.

[30] Jiaxin Chen, Xiaolin Tang, and Kai Yang, "A unified benchmark for deep reinforcement learning-based energy management: Novel training ideas with the unweighted reward," Energy, vol. 307, pp. 132687, 2024.