Scholar Publishing Group
International Journal of Social Sciences and Economic Management
https://doi.org/ 10.38007/IJSSEM.2025.060217
ISSN 2789-715X Vol. 6, Issue 2: 154-161

Construction and Optimization of Investment Decision Support System for Risk Management

Chenyang An

Queen Mary University of London, Economics and Finance, London, E1 4NS, United Kindom

Keywords: Investment Decision Support System; Risk management; Computer architecture; artificial intelligence

Abstract: With the increasing complexity of financial markets, the role of investment decision support systems in risk management is becoming increasingly important. This article explores the construction and optimization of investment decision support systems, with a focus on the system architecture of computer technology, including the framework application and analysis of databases, cloud data, and big data. The main focus is on data analysis input processing and storage, data analysis processing, and decision-making. In the field of risk prediction using artificial intelligence and machine learning, real-time processing and stream computing techniques, as well as optimization of decision support modules, were explored to achieve faster response and feedback of decisions. Research has shown that on the basis of computing resource allocation and optimization, the response speed and decision accuracy of investment decision support systems can be effectively improved.

Introduction

In the complex and ever-changing financial market, risk management has become an increasingly important influencing factor. In the rapidly developing and complex financial market, there are still many difficulties in using traditional analysis and decision support systems to complete timely information processing and corresponding decision support. With the rapid development of computer technology, artificial intelligence, and big data technology, investment strategy assistance systems that utilize these new technologies can effectively solve such problems. Combining investment risk assessment models, real-time data processing, and intelligent decision-making recommendations with the aforementioned auxiliary systems can better improve the accuracy and effectiveness of investment strategies. However, the current auxiliary investment strategy assistance system still has limitations in real-time data, model authenticity, and effectiveness. Therefore, the purpose of this article is to study how to use advanced computer science technology to achieve the progressive development of auxiliary investment strategy assistance systems and improve their practical significance in risk management.

1. Computer architecture design of investment decision support system

1.1. Overview of System Architecture and Technical Selection

In the process of building an investment decision support system, in order to meet the needs of practical applications and technological development, it is usually divided into several stages such as data collection, storage, processing, and decision result output, in order to effectively transmit and process the data.

For the selected solution, relational databases (such as MySQL or PostgreSQL) are often used as the core component for storing formatted data, while non formatted data is processed by non relational databases (such as MongoDB or Cassandra); By utilizing cloud computing technology, resources can be flexibly controlled to meet the demands of large-scale concurrent operations and massive data operations; Allocate and manage computing and storage resources on cloud servers such as Amazon Web Services, Microsoft Azure, or Google Cloud Platform to meet different needs; Big data architectures such as Apache Hadoop and Apache Spark can provide powerful data processing capabilities for investment decision support systems, enabling both real-time data streams and large-scale data analysis. With the cooperation of multiple technologies, real-time data reading, processing, and parsing can be achieved, providing accurate data reference basis for decision-makers.

1.2. Key modules of the system

The core functional modules of the investment decision support system include three key parts: data input, processing and storage, and analysis and decision-making. The core functional modules of the investment decision support system are shown in Figure 1.

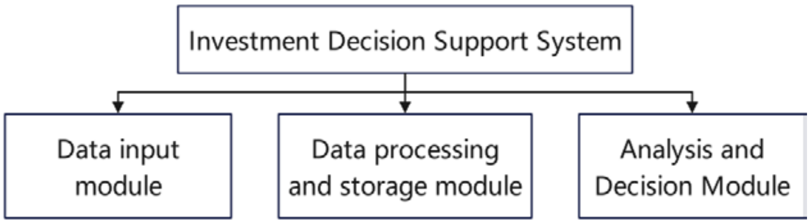

Figure 1. Core functional modules of investment decision support system

The main purpose of the data input module is to collect and store data in real-time from different sources (financial markets, company financial reports, news reports, etc.), ensuring the timeliness and accuracy of data information. This module can collect data in different formats, including structured data, unstructured data, and those in between. This module provides all the necessary information and materials for investors to make judgments.

The data processing and storage module cleans, converts, and preprocesses the collected data to ensure that the data quality meets the analysis requirements. Effectively store and manage data through database management systems (such as relational and non relational databases) for quick retrieval and analysis in the future.

Finally, using statistical analysis, artificial intelligence, and machine learning models, the processed data is deeply mined to provide investors with decision-making basis and assist them in making optimal investment decisions. This section includes decision-making methods such as regression, decision trees, neural networks, etc., to ensure the scientific and accurate completion of the decision-making process.

1.3. Core Technologies of System Design

By applying relevant core technologies, data cleaning and integration, and distributed computing provide technical support for the efficient operation of the system.

Data cleaning and integration is the first step in system data processing, mainly to improve data quality. The process includes cleaning redundant data, filling missing data, and filtering noisy data.

The commonly used data cleaning methods are based on filling the data with mean values and applying outlier detection methods to clean discrete data. Data integration refers to the process of integrating data from multiple channels into a standardized format for subsequent research work. The common formula for data integration is:

$$D_{\text{integrated}} = \bigcup_{i=1}^{n} D_i \tag{1}$$

Among them, Di represents the i-th data source, ∪ is the d
integrated dataset is the finalized dataset.

Distributed computing achieves greater computing power by jointly computing massive amounts of data on multiple computers, especially when dealing with massive amounts of information, which has incomparable advantages. In distributed computing such as Hadoop and Spark, parallel operations are implemented using the MapReduce algorithm. Tasks are allocated and executed by each computer in the Map, and then Reduce is responsible for aggregating the output data. The basic formula for MapReduce is:

$$\begin{aligned} &Map(k,v) \to (k',v') \\ &\mathrm{Re}\,duce(k,\{v'_1, v'_2, \ldots, v'n\}) \to (k', V_{result}) \end{aligned} \tag{2}$$

Among them, k is the key, v is the value, k' and v' are the mapped key value pairs, and the final output is merged in the Reduce stage. This mechanism can handle large-scale datasets and improve system efficiency.

2. Risk prediction model based on artificial intelligence and machine learning

2.1. Design and Selection of Risk Prediction Models

In investment decision support systems, risk prediction models are the core components that determine the system's ability to predict market volatility and potential risks. The design of risk prediction models requires selecting appropriate learning methods based on data characteristics and objectives. Supervised learning and unsupervised learning are two common learning methods.

Supervised learning uses annotated historical data for training, and the model learns the mapping relationship between input and output. Common supervised learning algorithms include regression analysis (such as linear regression, logistic regression) and classification algorithms (such as support vector machines, decision trees, random forests). Supervised learning is suitable for situations with clear target variables, such as predicting stock prices, risk levels, etc.

In contrast, unsupervised learning focuses more on exploring potential patterns and patterns in unlabeled data, such as clustering analysis (such as K-means, DBSCAN) and dimensionality reduction techniques (such as PCA principal component analysis). The most suitable scenario for exploring unlabeled data is to explore unknown and potential phenomena, such as the manifestation of unknown risks in the market and future predictions.

The data and purpose of adaptation determine which learning method to adopt. In practice, supervised learning and unsupervised learning are often used together to improve prediction accuracy and system adaptability.

2.2. Model Training and Evaluation

In the training process of risk prediction models, feature engineering and cross validation are key steps that help improve the predictive performance and stability of the model.

Feature engineering usually includes feature cleaning, feature selection, feature construction, feature normalization, etc. It is the process of transforming raw data into features that can represent the essence of the problem. Selecting features with high relevance and removing redundant or invalid features can reduce the complexity of the model and enhance its learning ability. For example, using standardization or normalization methods to compress feature values into the same range, to prevent certain features from affecting the model learning process too much or too little.

Cross validation is an important tool for evaluating model capability. The commonly used K-fold cross validation refers to dividing the data into K parts, using K-1 subsets for training each time, and repeating the remaining part as the test set K times to avoid overfitting and improve the applicability and breadth to new information. Mean precision is an effective tool for obtaining more accurate and complete evaluations of model capabilities.

3. Optimization of real-time data processing and decision support modules

3.1. Processing techniques for real-time data streams

Real time information flow control is crucial in decision support systems, and optimizing this process can greatly improve decision response speed and accuracy. To achieve efficient information flow control, Apache Kafka should be used as the information exchange tool. By increasing the number of theme partitions, data processing can be carried out simultaneously, which is beneficial for enhancing data flow and system scalability. By adjusting the batch configuration of the sending and receiving ends, frequent network access can be reduced, which can reduce latency and improve data processing efficiency. In addition, using compression techniques such as Snappy or Gzip can reduce the size of sent messages, combined with high-performance serialization techniques such as Avro, which can optimize the efficiency of the system. At the same time, Apache Flink is used for real-time data stream computation and analysis. It optimizes the handling of out of order data using an event timestamp mechanism, which allows data to be processed according to the actual time of the event rather than the time it was processed, preventing errors caused by time lag. At the same time, the optimization of window operations can be achieved through sliding or scrolling window techniques, which can reasonably divide the processing time of data, control the size of data groups, and optimize the use of computing resources, thereby reducing system burden. For event management, incremental snapshot and external storage methods (such as HDFS, RocksDB) can be used to store event data in a distributed manner to alleviate memory pressure in the system and ensure high-speed and stable handling of massive events. Through this method, the system is able to process live data streams with low latency and high traffic, ultimately providing accurate and fast information feedback to investors.

3.2. Optimization of Decision Support Algorithms

The main methods used for optimizing decision algorithms in investment decision support systems are multi-objective optimization and automatic decision-making. For multi-objective optimization problems, heuristic algorithms such as genetic algorithm (GA) or particle swarm optimization (PSO) are commonly used to solve multiple requirements. In operation, multiple optimization objective functions such as maximum return f1 (x) and minimum risk f2 (x) are defined and transformed into a single optimization problem. Taking genetic algorithm as an example, before the algorithm runs, a random population is generated, with each individual representing a possible decision solution. Then generate new individuals through selection, crossover, and mutation. Each generation selects individuals with high fitness for crossover and mutation to update the population until satisfactory results are achieved. In this process, fitness is achieved by iteratively finding the optimal solution layer by layer based on the weighted sum of objective function values or Pareto front.

Decision automation is often achieved through reinforcement learning (RL), which is based on the interaction between states and behaviors. In different states, different actions at are selected through the reward function R (st, at) for state transition. After each transition, adjustments are made according to different reward strategies. To achieve optimization, Q-learning algorithm is used to estimate the value Q (st, at) of each state action pair, and the strategy is adjusted through the following update formula:

$$Q(s_t, a_t) = (1-\alpha)Q(s_t, a_t) + \alpha\left[r_t + \gamma \max_{at+1} Q(s_t + 1, a_t + 1)\right] \tag{3}$$

Among them, α is the learning rate, γ is the discount factor, and rt is the immediate reward. By constantly interacting with the environment, the system gradually learns the optimal strategy and achieves automation of decision-making.

By combining multi-objective optimization and automated decision-making, the investment decision support system can dynamically adjust decisions in real-time based on market data and preset goals, achieving the effect of optimizing investment returns and reducing risks.

3.3. Improvement of Data Response Speed and Decision Feedback Mechanism

To optimize data response speed and decision feedback mechanism, it is necessary to first accelerate the data processing flow. In terms of data response speed, data processing can be optimized through distributed computing frameworks such as Apache Spark. Specifically, by distributing computing tasks across multiple nodes for parallel processing and using Spark RDD (Elastic Distributed Dataset) for parallel computing and storage of big data, computation time is significantly reduced. By storing data in HDFS (Hadoop Distributed File System) in a distributed manner, the reading speed of data can be effectively improved, avoiding the bottleneck problem of a single storage point. In addition, stream processing technologies such as Apache Kafka and Flink ensure fast transmission and processing of real-time data. When a new data source enters, the system can immediately trigger data processing tasks, reducing decision latency.

4. Performance evaluation of investment decision support system

4.1. System performance indicators

The main indicators for evaluating the effectiveness of investment decisions in support systems are response time, throughput, and concurrency capability. Response time refers to how long it takes for the system to complete user requests and respond, throughput refers to the number of transactions that the system can handle within a specified time unit, and concurrency capability refers to its ability to handle multiple demands.

Taking a certain investment management company as an example, during the peak period of the initial system, the response lag exceeded the standard by 200 milliseconds, resulting in poor customer experience and affecting the effectiveness and efficiency of real-time investment decision-making. In the improvement construction, ApacheSpark, a distributed computing architecture, was adopted to implement multi-threaded data processing, optimizing the problem of data processing time. The response speed reached 80 milliseconds, further improving the system's real-time performance.

In terms of efficiency, the initial efficiency of this system is QPS500. By clicking on the column storage of House and optimizing the database query, the efficiency of QPS 1500 can be achieved. At the same time, parallel query strategies are added to optimize the process and avoid query bottlenecks.

The initial system only supported synchronous processing of 50 synchronous requests. As the number of users increased, it may cause system crashes or lead to longer response times. To avoid this situation, Kubernetes is introduced to implement container deployment, allowing nodes to automatically expand, ensuring stable system operation even in large-scale synchronization situations, ultimately achieving the processing of over 200 synchronization requests. The following Figure 2 shows the comparison data of system performance before and after optimization.

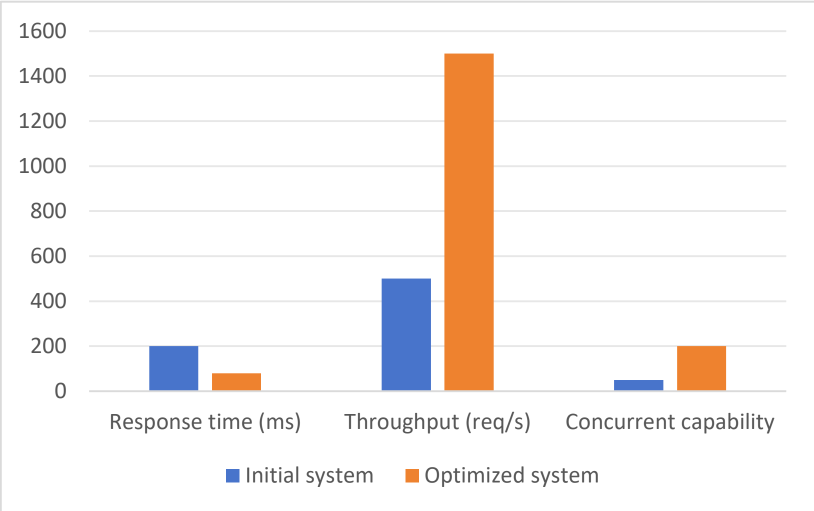

Figure 2. Comparison data of system performance before and after optimization

Through the above optimization, the response time, throughput, and concurrency of the system have been significantly improved, enabling the investment decision support system to operate efficiently during high-frequency trading and market fluctuations, greatly enhancing the practicality and decision support capabilities of the system.

4.2. Management and Optimization of Computing Resources

In the management and optimization of computing resources in investment decision support systems, load balancing and fault tolerance mechanisms are key to ensuring high availability and stability of the system. Load balancing avoids single point bottlenecks and overload issues by evenly distributing user requests across multiple servers. The fault tolerance mechanism ensures stable operation of the system in the event of hardware or computing node failures through redundant design and automatic recovery functions, without affecting decision support functions.

To achieve load balancing, load balancing software such as Nginx or HAProxy is used to evenly distribute the load. When the demand and server load reach a certain level, task forwarding is dynamically configured to ensure load balancing for each node, thereby improving the overall system processing rate and response speed. At the same time, distributed databases (such as Cassandra) and cloud computing environments (such as AWSEC2) are used to achieve backup and multiple backups for each node. If a node fails, it can be automatically transferred to a healthy node for operation, avoiding the occurrence of single point of failure.

After improvement, it can better handle batch requests, unexpected situations, and large-scale requests. The following table shows the system performance comparison of load balancing and fault-tolerant mechanisms before and after optimization.

Table 1. System performance comparison of load balancing and fault tolerance mechanisms before and after optimization

Performance index	Initial system	Optimized system
Request for balance	1:10	1:3
System availability	95%	99.9%
Fault tolerant recovery time	30s	5s

Through these optimizations, the load capacity and fault recovery capability of the system have been significantly enhanced, further improving the stability and efficiency of the investment decision support system.

5. Conclusion

This article focuses on the construction and optimization of investment decision support systems, exploring the application of key technologies such as computer architecture design, risk prediction using artificial intelligence and machine learning, real-time data processing, and optimization of decision support algorithms. Through system performance evaluation and feedback, it has been verified that optimization measures significantly improve the decision-making efficiency, return on investment, and user satisfaction of the system. In the future, with the further development of technology, investment decision support systems will continue to improve to cope with more complex market environments and enhance the scientificity and accuracy of decision-making.

References

[1] Buzdugan A, Capatana G. Advances in Developing a Decision Support System for Cyber Risk Management. Springer, Cham, 2024.

[2] Saskia H, Nijman S W J, Inge V, et al. The use of imputation in clinical decision support systems: a cardiovascular risk management pilot vignette study among clinicians. European Heart Journal - Digital Health, 2024(5):5.

[3] Stefano B, Simone B, Filippo B. The Decision Trees Method to Support the Choice of Economic Evaluation Procedure: The Case of Protection Forests. Forest Science, 2023(3):3.

[4] Gangadhari R K, Rabiee M, Khanzode V, et al. From unstructured accident reports to a hybrid decision support system for occupational risk management: The consensus converging approach. Journal of Safety Research, 2024, 89:91-104.

[5] Antonov A E, Buic G, Beiu C, et al. Functional system for experimentation and testing of products for professional use - Professional risk management tool for workers performing industrial activities. MATEC Web of Conferences, 2024, 389(000):10.

[6] D. Shen, "Complex Pattern Recognition and Clinical Application of Artificial Intelligence in Medical Imaging Diagnosis, " 2025 International Conference on Intelligent Communication Networks and Computational Techniques (ICICNCT), Bidar, India, 2025, pp. 1-8.

[7] X. Liu, "Research on User Preference Modeling and Dynamic Evolution Based on Multimodal Sequence Data, " 2025 2nd International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS), Hassan, India, 2025, pp. 1-7.

[8] J. Huang, "Performance Evaluation Index System and Engineering Best Practice of Production-Level Time Series Machine Learning System, " 2025 International Conference on Intelligent Communication Networks and Computational Techniques (ICICNCT), Bidar, India,

[9] *W. Han, "Using Spark Streaming Technology to Drive the Real-Time Construction and Improvement of the Credit Rating System for Financial Customers, " 2025 International Conference on Intelligent Communication Networks and Computational Techniques (ICICNCT), Bidar, India, 2025, pp. 1-6.*

[10] *M. Zhang, "Research on Joint Optimization Algorithm for Image Enhancement and Denoising Based on the Combination of Deep Learning and Variational Models, " 2025 International Conference on Intelligent Communication Networks and Computational Techniques (ICICNCT), Bidar, India, 2025, pp. 1-5.*

[11] *Y. Zou, "Research on the Construction and Optimization Algorithm of Cybersecurity Knowledge Graphs Combining Open Information Extraction with Graph Convolutional Networks, " 2025 2nd International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS), Hassan, India, 2025, pp. 1-5.*

[12] *Q. Xu, "Implementation of Intelligent Chatbot Model for Social Media Based on the Combination of Retrieval and Generation, " 2025 2nd International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS), Hassan, India, 2025, pp. 1-7.*

[13] *B. Li, "Research on the Spatial Durbin Model Based on Big Data and Machine Learning for Predicting and Evaluating the Carbon Reduction Potential of Clean Energy, " 2025 International Conference on Intelligent Communication Networks and Computational Techniques (ICICNCT), Bidar, India, 2025, pp. 1-5.*

[14] *Y. Zhao, "Design and Financial Risk Control Application of Credit Scoring Card Model Based on XGBoost and CatBoost, " 2025 International Conference on Intelligent Communication Networks and Computational Techniques (ICICNCT), Bidar, India, 2025, pp. 1-5.*

[15] *K. Zhang, "Optimization and Performance Analysis of Personalized Sequence Recommendation Algorithm Based on Knowledge Graph and Long Short Term Memory Network, " 2025 2nd International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS), Hassan, India, 2025, pp. 1-6.*

[16] *Chen M. Research on Automated Risk Detection Methods in Machine Learning Integrating Privacy Computing. 2025.*

[17] *Ding, J. (2025). Intelligent Sensor and System Integration Optimization of Auto Drive System. International Journal of Engineering Advances, 2(3), 124-130.*

[18] *Mingjie Chen. (2025). Exploration of the Application of the LINDDUN Model in Privacy Protection for Electric Vehicle Users. Engineering Advances, 5(4), 160-165.*

[19] *Liu, X. (2025). Research on Real-Time User Feedback Acceleration Mechanism Based on Genai Chatbot. International Journal of Engineering Advances, 2(3), 109-116.*

[20] *Zhang, M. (2025). Research on Collaborative Development Mode of C# and Python in Medical Device Software Development. Journal of Computer, Signal, and System Research, 2(7), 25-32.*