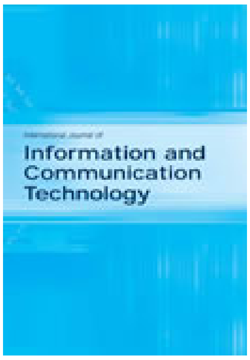
INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research



International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Construction of an economic data analysis and management system for legal regulation

Xiaohang Liang

DOI: 10.1504/IJICT.2025.10074948

Article History:

Received:	05 September 2025
Last revised:	11 October 2025
Accepted:	21 October 2025
Published online:	17 December 2025

Construction of an economic data analysis and management system for legal regulation

Xiaohang Liang

School of Law,
Shandong University of Finance and Economics,
Jinan, 250000, Shandong, China
Email: 13153178329@163.com

Abstract: This study addresses the efficiency limitations of traditional economic data analysis methods when processing large-scale, multi-source datasets under regulatory constraints. A blockchain-based system is proposed, featuring a novel sharding consensus algorithm with security and performance tradeoffs (SPTSCA) as its core component. Compared with existing approaches such as practical Byzantine fault tolerance (PBFT) and OmniLedger, the key innovations of SPTSCA include a dynamic shard adjustment mechanism for improved load balancing and an optimised consensus process that minimises communication rounds. Experimental results demonstrate that, due to more balanced shard formation, SPTSCA achieves up to a 1.49% increase in throughput compared with OmniLedger. More importantly, its performance significantly surpasses that of PBFT, with maximum throughput improvements of 143.1% and latency reductions of 89.1%. The algorithm enables secure large-scale economic data sharing, providing robust technical support for regulatory authorities.

Keywords: legal regulation; economic data analysis; management system; blockchain.

Reference to this paper should be made as follows: Liang, X. (2025) 'Construction of an economic data analysis and management system for legal regulation', *Int. J. Information and Communication Technology*, Vol. 26, No. 47, pp.53–70.

Biographical notes: Xiaohang Liang is currently a junior undergraduate student majoring in law at Shandong University of Finance and Economics, Jinan City, Shandong Province, China, having enrolled in 2023. She has been awarded the First-Class University Scholarship for two consecutive academic years (2023–2024 and 2024–2025).

1 Introduction

In an era marked by rapid globalisation and digital transformation, economic activities are growing exponentially in scale, complexity, and dynamism. Massive transactions by multinational corporations, the volatility of financial markets, and the rise of new digital economic models have created a vast and intricate economic network. For legal and regulatory authorities, the ability to track economic activity accurately, detect emerging risks promptly, and implement effective oversight measures has become essential. These capabilities are critical for maintaining market order, ensuring stable and sustainable socio-economic development, and safeguarding fairness and justice (Kalamara et al., 2022; Chukwuma-Eke et al., 2022; Gongada et al., 2024).

However, traditional methods of economic data analysis and management are increasingly unable to meet the demands of today's complex economic landscape. The explosive growth of data has outpaced the capacity of conventional statistical and analytical tools, which struggle to efficiently collect, process, and extract insights from massive datasets within a reasonable time. As a result, regulatory decisions often lag behind real-time developments. Moreover, the growing diversity of economic data – ranging from structured to semi-structured and unstructured formats – poses additional challenges, as traditional tools are ill-suited to integrating and analysing such heterogeneous information. Meanwhile, legal regulation faces mounting challenges. New forms of economic crime continue to emerge, marked by greater concealment, complexity, and cross-border operations. These crimes frequently exploit complex financial instruments, virtual currency transactions, and multinational corporate structures to evade traditional oversight mechanisms (Bhuiyan et al., 2022; Li et al., 2023; Kumbure et al., 2022). In the realm of financial technology, activities such as illegal fundraising and money laundering have rapidly expanded by leveraging the anonymity of internet platforms and virtual assets, making them particularly difficult to detect and trace (Tulli, 2023; Lehmann, 2023). These developments not only threaten national economic security but also place unprecedented pressure on the effectiveness of legal supervision.

To address these challenges, big data analytics has found extensive applications in economic monitoring, financial risk early warning, and market behaviour analysis. This technology can process massive, heterogeneous information and identify latent patterns and anomalies within complex economic activities. However, the current application of big data analytics in the economic domain still faces several critical bottlenecks. These include trust deficiencies caused by diverse data sources, low efficiency in cross-system data sharing and collaborative analysis, and significant analysis delays in regulatory scenarios that require high real-time performance. These limitations constrain the ability of regulatory authorities to gain timely insights and intervene in high-risk activities.

This study proposes the use of neural network algorithms to automatically extract discriminative features from large-scale, multi-source economic data. By mining multidimensional inputs – including corporate financial records, market transaction data, and online public opinion – the system aims to identify indicators of economic crime, patterns of market manipulation, and corporate compliance risks. This enables regulatory bodies to more precisely identify high-risk entities and activities. Through a deep integration of economic data mining techniques and blockchain technology, the study develops an efficient, secure, and scalable system architecture. The system specifically targets key issues in legal supervision, such as anti-money laundering, market manipulation detection, and corporate compliance risk warning. It is designed to automatically extract discriminative features from large-scale, multi-source economic data, enabling regulatory authorities to more accurately identify high-risk entities and activities. For example, the system can mine potential risk signals from corporate financial records, market transaction data, and online public opinion, thereby improving the detection of concealed economic crimes, supporting real-time regulatory decision-making, and enhancing the precision and effectiveness of legal oversight. These applications highlight the system's practical value in maintaining market order and promoting sustainable economic development.

2 Related work

Over the past decade, big data and related technologies have gained rapid momentum. Although their application in macro-finance remains in its early stages, the field is evolving quickly. Data mining technologies provide regulatory agencies with in-depth, 'penetrative' analytical capabilities. These tools can extract potential risk signals from complex economic activities, supporting more accurate and timely decision-making. Salisu et al. (2022) noted that data mining uncovered hidden patterns within massive and often incomplete economic datasets, revealing relationships that traditional statistical methods struggled to detect. For instance, association rule mining can identify abnormal transaction patterns between enterprises, aiding antitrust investigations. Classification algorithms are widely used in credit risk assessment and financial fraud detection. Visser et al. (2022) further demonstrated the effectiveness of the data mining tool FineDataLink in data integration and cleaning. It addressed challenges related to poor data quality and high heterogeneity, thus improving the reliability of analytical results.

In time series analysis and economic forecasting, Ghauri et al. (2020) highlighted the broad use of the autoregressive integrated moving average (ARIMA) model for predicting macroeconomic indicators. By capturing cyclical patterns in historical data, time series models help regulatory agencies identify economic trends and inform policy development. ARIMA also improves forecasting accuracy by differencing non-stationary data, effectively addressing common issues like trend and seasonality. Kim (2022) introduced a new time series model for short-and medium-term economic forecasting. This model integrated Fourier series with an ARMA (n, n-1) structure. It first removed long-term trends using curve fitting, then analysed seasonal variation through Fourier components, and finally modelled irregular fluctuations with ARMA (n, n-1). Compared to the traditional ARMA(p, q) model, this approach simplified the structure and improved forecasting performance.

Zhu et al. (2021) examined how blockchain technology influenced the quality of corporate financial reporting. Their empirical findings showed that blockchain-based invoicing enhanced financial transparency, leading to outcomes such as increased stock liquidity, reduced dispersion in analyst forecasts, and a lower cost of equity capital. Zheng et al. (2023) proposed a blockchain-based traceability framework for sharing personal financial data. Utilising smart contracts, the system acts as a trusted intermediary between users and third-party platforms. It offers transparent validation, privacy protection, and traceable provenance – features that align with the strict authentication and traceability requirements of open banking environments.

Moreover, big data analytics has demonstrated significant potential in enhancing the resilience and decision-making quality of economic systems. Jiang et al. (2024) showed that the synergy between supply chain integration and big data analytics capabilities was essential for building supply chain resilience, though its effectiveness depends heavily on a foundation of high-quality data. At the level of small and medium-sized enterprises, Mehmood et al. (2025) found that big data analytics could simultaneously improve economic and environmental performance by promoting green innovation; however, achieving this transformation imposes stringent requirements on the breadth and reliability of available data. Research in the field of financial decision-making further supports this point. Al-Okaily and Al-Okaily (2025) emphasised that data quality, analytical capability, and system integration were the key factors influencing the quality of data-driven financial decisions. Similarly, Kumar et al. (2024), in their systematic review on supply chain decarbonisation, emphasised that big data analytics was a key enabling technology. However, its effective implementation is still limited by several factors, including restricted data accessibility, barriers to inter-organisational collaboration, and the absence of standardised frameworks. These challenges also extend to the field of management accounting. Abdelhalim (2024) revealed that the integration of management accounting practices with big data analytics could effectively enhance corporate sustainability, but the entire process relied heavily on reliable and verifiable data inputs. In summary, while existing research affirms the value of big data analytics, it also consistently points to deep-seated bottlenecks such as data credibility, cross-system sharing inefficiencies, and governance challenges. This underscores the urgent need to build a new type of infrastructure that ensures data authenticity, transparency, and efficient circulation – thereby providing a clear theoretical and practical foundation for the blockchain-based economic data management and analysis system proposed in this study.

Recent research has made notable progress in applying blockchain technology to economic data analysis. However, most existing studies emphasise its theoretical advantages while overlooking its real-world impact. Specifically, there has been limited examination of how blockchain can improve data trustworthiness and sharing efficiency in practical settings. This study addresses that gap by exploring how to design a secure and high-performance blockchain-based framework for economic data analysis from a legal and regulatory perspective. The goal is to enable trustworthy data sharing and deeper insight into economic behaviour.

3 Method

3.1 Economic data mining and analysis methods

In economics, data are a fundamental object of analysis and come in diverse forms, each with unique characteristics. Extracting their full value often requires specialised analytical approaches tailored to the data type. At the macroeconomic level, quantitative data typically include indicators such as gross domestic product (GDP), inflation, unemployment rates, and interest rates. In contrast, qualitative data – such as information on economic policy types or industrial policy directions – play a different role. Government measures, including expansionary fiscal policy or contractionary monetary policy, exert wide-ranging impacts on economic performance. Industrial policies, in particular, guide resource allocation toward key sectors, supporting structural transformation and industrial upgrading. These policies are generally developed through qualitative assessments and strategic planning.

Compared to traditional datasets, big data are defined primarily by their immense volume. While conventional data are measured in bits or megabytes, big data typically span terabytes (TB) or even petabytes (PB). In addition to size, big data are highly varied in form and require the ability to process both structured and unstructured content. As a comprehensive problem-solving approach, big data mining facilitates the acquisition, storage, processing, and application of data resources. Its primary goal is to extract meaningful insights from vast, complex datasets. Compared with traditional data analysis, big data mining differs significantly in its targets, scope, and objectives, as illustrated in Figure 1.

Figure 1 Comparison between traditional data analysis and big data mining (see online version for colours)

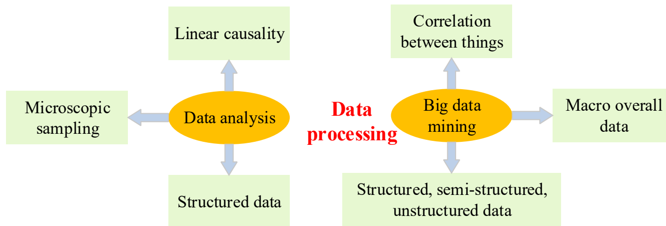
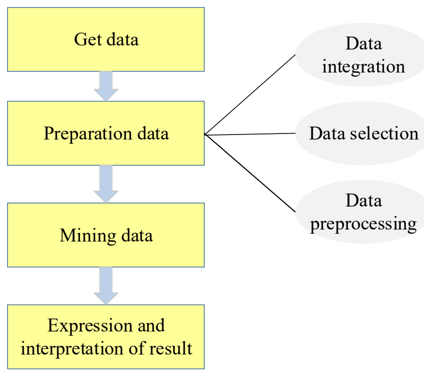


Figure 2 Data mining process flow (see online version for colours)



Data mining is the process of uncovering inherent patterns and valuable insights from large volumes of seemingly unstructured and irregular data. It typically combines statistical software with modern computing technologies to extract meaningful and actionable information from massive datasets, facilitating informed decision-making in various domains. However, as data volumes grow excessively large – often reaching terabyte or petabyte scales – the efficiency of traditional mining algorithms tends to decline due to computational and memory constraints. In such scenarios, scalable infrastructure such as cloud computing platforms becomes essential for handling data storage, processing, and real-time analytics. These platforms offer distributed computing capabilities and elastic resources that help overcome the limitations of conventional data analysis tools. Data mining typically consists of four key stages. The first is data acquisition, where raw data are gathered from various sources. The second is data preparation, involving cleaning, integration, and transformation of the data. The third stage is data mining, where analytical methods are applied to uncover patterns. Finally, result interpretation involves evaluating and visualising the findings for practical use (Korinek, 2023). These stages are illustrated in Figure 2 and serve as a foundational framework for conducting structured and efficient data analysis.

Association rule mining seeks to identify frequent co-occurrence relationships between itemsets in a dataset. It is commonly used in applications like market basket analysis. For example, if customers who buy item A are also likely to purchase item B, this information can guide product recommendations and marketing strategies to boost cross-selling and customer retention. The Apriori algorithm is a classic method for mining such association rules. It finds frequent itemsets and generates rules based on minimum support and confidence thresholds, effectively filtering out statistically insignificant patterns. In economic data analysis, association rule mining can uncover correlations between different economic indicators. For instance, it can identify patterns linking inflation and unemployment rates over specific periods, helping policymakers monitor macroeconomic dynamics. Time series analysis examines data points collected sequentially over time to detect trends, seasonality, and cycles, and to forecast future values. This technique is widely applied in economics, including stock price forecasting, sales prediction, GDP growth estimation, and other forms of macroeconomic indicator analysis (Li et al., 2023; Lee and Mangalaraj, 2022; Tiozzo Pezzoli and Tosetti, 2022). Table 1 summarises key data mining methods, their objectives, common algorithms, and typical application scenarios.

Table 1 Comparison of data mining methods

Data mining method	*Primary objective*	*Common algorithms*	*Application scenarios*
Classification and prediction	Predict the category or value of new observations based on known labels.	Decision trees, neural networks, logistic regression	Credit scoring, stock price forecasting, consumer behaviour prediction.
Clustering analysis	Group data objects into clusters to reveal underlying structure.	K-means, hierarchical clustering	Market segmentation, industry analysis, customer profiling.
Association rule mining	Discover frequent co-occurrence relationships among items.	Apriori algorithm	Market basket analysis, economic indicator correlation analysis.
Time series analysis	Identify temporal patterns and forecast future values.	ARIMA model	Stock market forecasting, sales forecasting, macroeconomic predictions.
Anomaly detection	Identify abnormal data points within a dataset	Statistical, clustering-based, classification-based methods.	Fraud detection, data quality monitoring, abnormal economic event identification.

Clustering analysis aims to reveal inherent patterns within large datasets by grouping data based on shared characteristics. It organises data into distinct clusters and often presents the results visually; using charts or tables, to help users better understand underlying structures. Unlike classification methods, clustering is especially useful when dealing with large datasets that lack predefined categories. Given a dataset *M*, the objective is to partition it into *x* clusters based on selected features. Various clustering algorithms group samples with similar characteristics into specific clusters, ensuring that each data point belongs to exactly one cluster. The resulting organisation must satisfy the following conditions equation (1):

$$\begin{Bmatrix} M_1 \cup M_2 \cup M_3 \ldots \cup M_X = M \\ M_i \cup M_j = \varnothing \end{Bmatrix} \tag{1}$$

After performing clustering analysis on a text dataset, the entire sample set can be divided into multiple subclasses based on user requirements and certain feature conditions. Subsequently, an unsupervised classification of the sample data can be completed, which is valuable for implementing personalised recommendations based on the classification results. In clustering analysis, to determine the similarity between different data points, it is first necessary to define the clustering statistic. Once this quantitative metric is established, quantitative methods can be applied for clustering analysis. Suppose there are n variable objects and p objects in an n-dimensional space. Each object exists as a point in this space, and the distance between points reflects the similarity between objects. If two n-dimensional vectors are $x_i = (x_{i1}, x_{i2}, \ldots, x_{in})$ and $x_j = (x_{j1}, x_{j2}, \ldots, x_{jn})$, there are various ways to measure the dissimilarity between objects, including Minkowski distance, Manhattan distance, Euclidean distance, and Chebyshev distance. These are respectively expressed in equations (2)–(5):

$$d_q\left(x_i, x_j\right) = \left\|x_i - x_j\right\|_q = \left(\sum_{k=1}^{n}\left|x_{ik} - x_{jk}\right|^q\right)^{\frac{1}{q}} \tag{2}$$

$$d_1\left(x_i, x_j\right) = \left\|x_i - x_j\right\|_1 = \left(\sum_{k=1}^{n}\left|x_{ik} - x_{jk}\right|\right) \tag{3}$$

$$d_2\left(x_i, x_j\right) = \left\|x_i - x_j\right\|_2 = \left(\sum_{k=1}^{n}\left|x_{ik} - x_{jk}\right|^2\right)^{\frac{1}{2}} \tag{4}$$

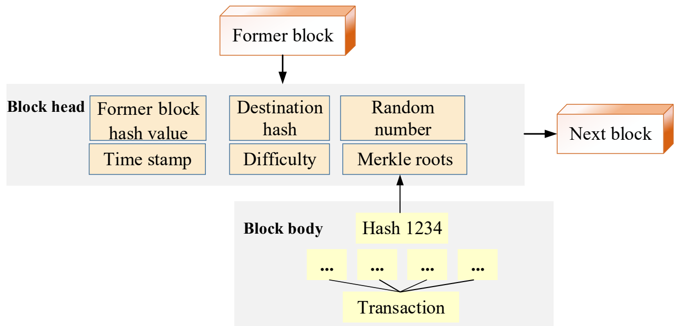
$$d_\infty\left(x_i, x_j\right) = \left\|x_i - x_j\right\|_\infty = \max_{k\in(1,2,\ldots,n)}\left|x_{ik} - x_{jk}\right| \tag{5}$$

Equation (2) generalises the distance metric with parameter $q \in [1, \infty)$. When q equals 1, 2 or ∞, the Minkowski distance between objects corresponds to the Manhattan distance (3) or Euclidean distance (4), respectively, with Chebyshev distance shown in equation (5).

3.2 Blockchain technology and principles

Blockchain is essentially a decentralised distributed database that records all transactions since the network's inception. Authorised parties can access these records. The blockchain consists of two main parts: the block header and the block body, as shown in Figure 3. The block header contains essential information such as the current block's hash value, timestamp, and the Merkle tree root, which is used to efficiently and securely summarise all transactions in the block. The block body stores multiple transaction records bundled within the block. The blockchain network is maintained by distributed nodes without a central authority. Each node stores a complete copy of the ledger and communicates through a peer-to-peer network for data transmission and verification. This decentralised design provides the system with high fault tolerance and strong resistance to attacks. Once a transaction is recorded on the blockchain, it becomes immutable – it cannot be altered or deleted. This permanence is ensured because each block includes the hash of the previous block. Any attempt to modify a block changes its hash and all subsequent hashes, which other nodes in the network can easily detect and reject as invalid (Wang et al., 2022; Jabeur et al., 2024). This design reinforces trust and data integrity across the entire network.

Figure 3 Block structure (see online version for colours)



The blockchain hash algorithm is expressed by equation (6):

$$H_{block} = H\left(H_{prev}, Data, Nonce\right) \tag{6}$$

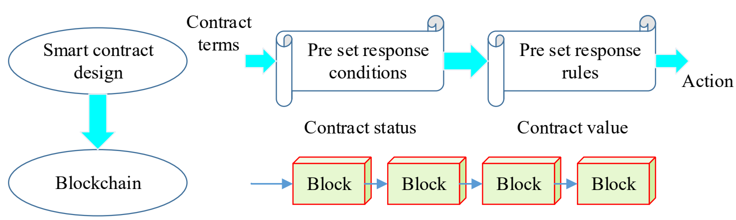
The block hash value H_{block} is generated from the previous block's hash H_{prev}, the current block's transaction *Data*, and a *Nonce*.

The difficulty target of the proof-of-work (PoW) can be represented by equation (7):

$$W = 2^k \tag{7}$$

where k denotes the difficulty parameter, and W specifies the required number of leading zeros that the block hash must satisfy.

Figure 4 Operational flow of smart contracts (see online version for colours)



Smart contracts are a crucial part of blockchain technology. They are computer programs that run on the blockchain and automatically enforce contract terms without requiring third-party intermediaries (Sabirov and Abduvaliyeva, 2022; Bakır et al., 2022; Rao et al., 2023). By encoding contract terms into code, smart contracts execute automatically when specified conditions are met, enhancing both transaction efficiency and transparency. When active, a smart contract receives external input data alongside contract conditions. Once these inputs fulfil the preset conditions, the contract triggers a series of actions – from action 1 through action N – based on predefined rules. As these actions execute, the contract's state and values are updated accordingly. These updates are then recorded in a new blockchain block. This block contains the latest contract state, its updated values, and all relevant transaction details. It is added to the end of the blockchain, guaranteeing the immutability and permanence of the data. This automated mechanism eliminates manual intervention, reduces execution time, and minimises the risk of errors or fraud, making smart contracts highly applicable in financial, legal, and supply chain contexts. Figure 4 illustrates the operational flow of smart contracts.

The state transition of a smart contract can be expressed by equation (8):

$$Y(S,T)=S' \tag{8}$$

where S represents the previous state of the smart contract, T denotes the set of input transactions, and S' is the new state after executing the transactions.

The execution logic of the smart contract is given by equation (9):

$$C=f(P,T,S) \tag{9}$$

P refers to contract writing, T to contract deployment, and S to contract execution.

The digital signature algorithm is equation (10):

$$Sign(m,s)=(r,s) \tag{10}$$

This equation indicates that the private key s is used to sign message m, generating the signature (r, s).

The algorithm *KeyPair* generates the public key pk and private key sk as equation (11):

$$KeyPair(n)=(pk,sk) \tag{11}$$

The automated execution of smart contracts can be represented as equation (12):

$$S_{t+1}=S_t+f\left(S_t,T_t\right) \tag{12}$$

where S_t is the contract state at time t, and T_t is the external input at time t.

3.3 Trusted economic data sharing and management system

This section focuses on constructing a trusted economic data sharing and management system. It innovatively proposes the sharding consensus algorithm with security and performance tradeoffs (SPTSCA) to significantly enhance system performance while ensuring data security. SPTSCA operates in epochs, each divided into two parts: sharding and consensus. This design guarantees both efficient system operation and data security and consistency across the network. The sharding component is responsible for logically partitioning the blockchain network into multiple shards based on resource availability and node capabilities, enabling parallel transaction processing. The consensus component ensures the validity of transactions within each shard and maintains overall network state consistency through coordinated validation mechanisms. This dual-phase structure effectively balances scalability, reliability, and fault tolerance. The operational cycle of SPTSCA is illustrated in Figure 5.

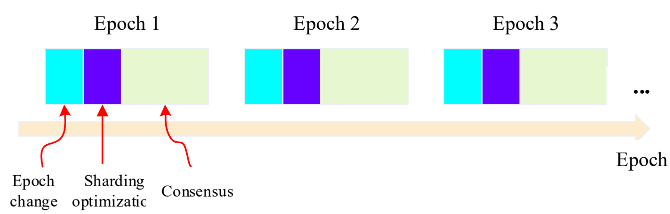
During the sharding phase, SPTSCA first determines the optimal shard size based on network conditions and historical data. This process considers factors such as node computing power, network bandwidth, and storage capacity to ensure optimal resource allocation. By analysing real-time network status, the algorithm dynamically adjusts the number and size of shards to maintain balanced processing capacity across all shards.

When new nodes seek to join the blockchain network, they send requests to existing nodes containing their identity, resource status, and proof of capability. The existing nodes rigorously verify the new node's legitimacy, reputation, and capabilities. Once approved, the new node selects the most suitable shard based on its resources and network guidance, then submits a join request to that shard. The shard's nodes vote on the request, and if it gains sufficient support, the new node officially joins the shard and participates in transaction processing and consensus. During node migration, nodes are ranked by resource capacity and reputation. High-resource, high-reputation nodes are prioritised for migration to shards with lighter workloads. The migration cost function $C_{migrate}$ measures the cost of node migration, including data transmission and state synchronisation overhead equation (13):

$$C_{migrate} = \alpha \times DataSize + \beta \times SyncTime \tag{13}$$

where $DataSize$ is the amount of data to be migrated, $SyncTime$ is the state synchronisation time, and α and β are weighting coefficients.

Figure 5 Operational cycle of SPTSCA (see online version for colours)



To strictly maintain data consistency during node migration and dynamic shard reconfiguration, the system employs a state snapshot and incremental synchronisation mechanism. When a shard requires restructuring, the source shard first generates a deterministic state snapshot that captures all pending, non-finalised transactions. Before a migrating node leaves its original shard, it synchronises the intermediate state data – based on this snapshot – to the target shard. During this process, the system temporarily activates a global lock mechanism, which pauses the processing of new transactions within the affected data range to ensure snapshot stability. Once synchronisation is complete, nodes in the target shard execute a cross-validation procedure using the same verification function to confirm the accuracy and integrity of the received state data. Only after successful verification does the migrating node officially participate in consensus within the target shard, at which point the global lock is released and the system resumes normal operation. This strategy effectively prevents data inconsistency or double-spending issues that may arise from node movement, ensuring global consistency throughout shard reorganisation.

To further clarify the dynamic shard adjustment strategy, Algorithm 1 outlines the procedure for periodically optimising shard configuration based on real-time system load and node performance metrics.

In the consensus phase, SPTSCA utilises an innovative, scalable reconfiguration consensus mechanism. Each shard runs an optimised consensus protocol involving three stages: pre-prepare, prepare, and commit. The proposing node first creates a block proposal and broadcasts it to the shard's nodes. Upon receiving the proposal, nodes verify it and, if valid, send out pre-prepare messages. Once enough pre-prepare messages are collected, nodes move to the prepare phase and broadcast prepare messages. After gathering sufficient prepare messages, the system advances to the commit phase, where commit messages are broadcast, finalising the transaction and updating the local state.

Algorithm 1 Dynamic shard adjustment algorithm

Input: Current shard set S, total system transaction load L, node performance metric set P

Output: Updated shard set S'

For each shard s_i in S:

Compute shard load ratio: $l_i = \dfrac{s_i \cdot transation_{count}}{L}$

Compute average node performance: p_i = average(s_i.nodes.performance)

End for

if any l_i > load threshold θ_1 or p_i < performance threshold θ_p:

Perform shard splitting: divide the overloaded or low-performance shard s_i evenly based on node performance

Else if any l_i < merge threshold θ_m and $|S| > 1$:

Perform shard merging: merge the underloaded shard s_i with an adjacent shard s_j

End if

Return the updated shard set S′

The internal workflow of the scalable reconfigurable consensus mechanism operates within each shard and consists of three stages, as shown in Algorithm 2.

Algorithm 2 Scalable reconfigurable consensus mechanism (within a single shard)

Input: Transaction proposal T_x, current shard node list N

Output: Consensus result (Commit or discard)

// Stage 1: Pre-prepare

The proposer constructs a block B and broadcasts a pre-prepare message ⟨PRE-PREPARE, B⟩ to N.

Upon receiving the message, each node n_i:

Verifies the validity of B (e.g., signature, format).

If valid, broadcasts a prepare message ⟨PREPARE, B, n_i⟩.

// Stage 2: Prepare

Each node n_i collects prepare messages.

When more than 2f valid prepare messages are received (where f is the maximum number of tolerated faulty nodes):

Broadcast a commit message ⟨COMMIT, B, n_i⟩.

// Stage 3: Commit

Each node n_i collects commit messages.

When more than 2f valid commit messages are received:

Commit block B to the local blockchain and update the local state.

If any stage times out before reaching the threshold, discard proposal B.

As the number of network nodes and transaction volumes change, the system periodically reorganises and optimises shards based on predefined dynamic adjustment policies. During this process, some nodes migrate between shards following specific rules, allowing shards to split or merge as needed. If a shard becomes too large and performance declines, it splits into smaller shards. Conversely, shards with light transaction loads are merged to minimise resource waste.

3.4 Experimental validation

The financial stock dataset from Alibaba Cloud's MaxCompute platform was chosen to experimentally validate the performance advantages of the proposed SPTSCA method in economic data analysis. MaxCompute encompasses a wide range of stock-related data, including historical prices, trading volumes, and financial indicators, enabling the simulation of diverse scenarios relevant to economic behaviour and regulatory analysis. This dataset is stored within the public project *BIGDATA_PUBLIC_DATASET* on MaxCompute and can be accessed via the corresponding schema name, making it convenient for reproducibility and scalability in experimental research.

To evaluate the proposed approach, a small blockchain network was deployed using multiple high-performance servers with the following hardware specifications: Intel Xeon Gold 6248 CPUs and 128 GB DDR4 2933 MHz RAM, ensuring sufficient computational power for high-throughput transaction processing. The blockchain environment was built on the Hyperledger Fabric framework, chosen for its modular architecture and support for permissioned networks. The SPTSCA algorithm, along with the relevant smart contract logic, was implemented using the Go programming language, which is natively supported by Hyperledger Fabric. The system operated under Ubuntu Server 20.04 LTS, providing a stable and secure Linux environment suitable for enterprise-grade distributed applications.

4 Results and discussion

4.1 Evaluation of SPTSCA latency and throughput

The throughput and latency of SPTSCA, practical Byzantine fault tolerance (PBFT), and OmniLedger were tested by varying the number of nodes, as shown in Figures 6 and 7. The experiments used between 4 and 48 nodes, increasing in steps of 4, with 5% of nodes designated as malicious. As the number of nodes grew, both SPTSCA and OmniLedger maintained a stable throughput of about 2,000 transactions per second (tx/s). In contrast, PBFT's throughput dropped sharply. This decline is due to PBFT's consensus process relying on full network broadcasting. As more nodes join, communication overhead rises, reducing transaction efficiency. SPTSCA and OmniLedger address this by using sharding to process transactions in parallel, allowing Byzantine consensus algorithms to scale effectively. With the same number of nodes, SPTSCA showed a slight throughput improvement over OmniLedger, peaking at a 1.49% increase. This gain results from SPTSCA's optimised shard formation, which balances shard distribution more evenly. OmniLedger's random sharding, by contrast, ignores node performance differences, potentially causing bottlenecks in some shards. Regarding latency, when node counts are low, SPTSCA experiences the highest latency, followed by OmniLedger, with PBFT performing best. This is because SPTSCA's dynamic sharding and optimisation require extra time to build effective shard sets. Overall, in the tested environment, SPTSCA achieved up to a 143.1% increase in throughput and reduced latency by as much as 89.1% compared to PBFT. These results show that SPTSCA significantly outperforms PBFT in scalability and also holds a slight advantage over OmniLedger.

Figure 6 SPTSCA throughput evaluation (see online version for colours)

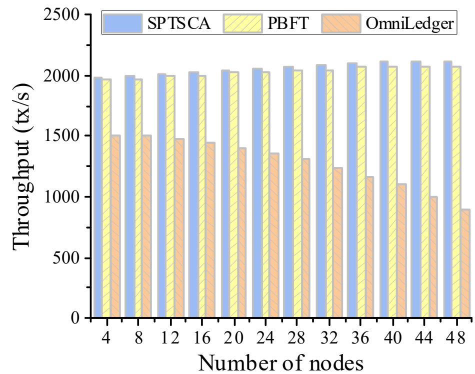
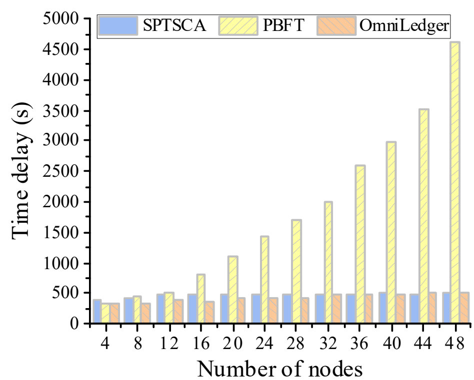


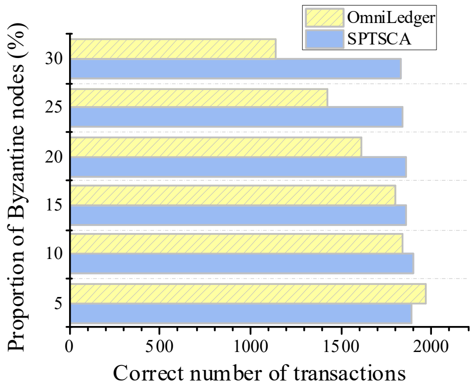
Figure 7 SPTSCA latency evaluation (see online version for colours)



4.2 SPTSCA fault tolerance evaluation

The total number of nodes was set at 150, with the proportion of malicious nodes ranging from 5% to 30%. A total of 2,000 transactions were submitted to evaluate the performance of both algorithms under varying levels of malicious activity and to test their fault tolerance capabilities. As the share of Byzantine nodes increased, OmniLedger experienced a significant drop in the number of correctly processed transactions. In contrast, SPTSCA maintained relatively stable performance and was less impacted by malicious behaviour. Further analysis showed that OmniLedger's random sharding leads to uneven distribution of Byzantine nodes across shards. When any shard contains more than one-third Byzantine nodes, its normal operation is severely compromised. This disruption causes transactions to be processed out of order, increases the likelihood of consensus delays, and may ultimately result in consensus failure or system instability. Figure 8 presents SPTSCA fault tolerance evaluation.

Figure 8 SPTSCA fault tolerance evaluation (see online version for colours)



To quantitatively verify the advantages of SPTSCA in shard load balancing, this study analysed the distribution of transaction processing across shards under a 48-node configuration. During the experiment, both SPTSCA and OmniLedger processed 10,000 transactions. The standard deviation of the load for each shard was calculated, as shown in Table 2. A smaller standard deviation indicates a more balanced load. The results showed that under the same total transaction volume, the load standard deviation of shards generated by SPTSCA was significantly lower than that of OmniLedger. This finding directly confirms that the dynamic sharding strategy of SPTSCA – based on node performance and resource status – effectively creates shard sets with more balanced load capacities. It also prevents the issue of individual shards becoming early performance bottlenecks, which may occur in OmniLedger's random sharding. Consequently, this balance provides a solid explanation for SPTSCA's stable throughput and slight performance improvement.

Table 2 Comparison of shard load balancing (48 nodes)

Algorithm	*Number of shards*	*Average load (tx/shard)*	*Load std. dev.*
SPTSCA	4	2,500	86.4
OmniLedger	4	2,500	342.7

4.3 Cold start latency decomposition

To investigate the reasons behind the higher latency of SPTSCA under a small number of nodes, the study conducted a fine-grained time breakdown of the consensus process during the cold start phase (16 nodes). The focus was on the additional overhead introduced by dynamic sharding and optimisation. Table 3 presents the time consumption comparison between SPTSCA and PBFT during their first consensus round.

Analysis of Table 3 shows that among the total latency of 91.1 ms for SPTSCA, shard formation alone accounted for 45.2 ms – nearly half of the total delay. In contrast, PBFT did not require this process. This result quantitatively confirms that the extra time mainly stemmed from the construction of the initial shard set, which involved computation-and communication-intensive tasks such as node resource evaluation, shard planning, and node allocation. This analysis points to future optimisation directions, such as accelerating startup through pre-configuration or lightweight sharding protocols, which is particularly critical for small-scale deployment scenarios.

Table 3 Cold start latency breakdown (ms)

Algorithm	*Shard formation*	*Pre-prepare*	*Prepare*	*Commit*	*Total latency*
SPTSCA	45.2	12.1	15.3	18.5	91.1
PBFT	N/A	10.5	13.8	16.2	40.5

4.4 Performance across different economic data scenarios

To evaluate the adaptability of the SPTSCA algorithm in various economic data analysis tasks, the study simulated three representative scenarios:

a High-frequency stock trading data.

b Macroeconomic indicator time-series data.

c Complex inter-firm supply chain transaction networks.

The experiments were conducted using 64 nodes, and the results are summarised in Table 4.

Table 4 Performance comparison across different economic data scenarios

Data scenario	*Algorithm*	*Avg. throughput (tx/s)*	*Avg. latency (ms)*	*CPU utilisation (%)*
Scenario A (high-frequency)	SPTSCA	21,50	89	78.5
	RapidChain	1,980	95	82.1
Scenario B (macroeconomic)	SPTSCA	2,080	92	72.3
	RapidChain	1,920	101	75.6
Scenario C (complex network)	SPTSCA	1,950	105	85.2
	RapidChain	1,750	128	88.7

From Table 4, it can be observed that across different economic data types and complexity levels, SPTSCA consistently outperformed the benchmark RapidChain in both throughput and latency. The advantage was most pronounced when processing the structurally complex and highly interconnected supply chain network data, where throughput improved by approximately *11.4%*, and latency decreased by about *18.0%*. These results indicate that SPTSCA's dynamic resource management mechanism effectively adapts to the processing demands imposed by varying data characteristics. Moreover, SPTSCA exhibited slightly lower CPU utilisation in all scenarios compared to RapidChain, suggesting more efficient resource usage achieved through balanced load distribution. Overall, the findings demonstrate SPTSCA's strong adaptability and robustness in handling diverse economic data analysis tasks.

4.5 Security analysis

Although the inherent characteristics of blockchain – such as immutability and Byzantine fault tolerance – form the foundation of system security, sharded blockchains face specific types of attacks that require dedicated countermeasures. This section discusses how SPTSCA addresses two major threats: Sybil attacks and shard-specific attacks. In a Sybil attack, an adversary attempts to create numerous fake identities to gain control over the network. SPTSCA mitigates this threat through a strict node admission mechanism. When a new node joins, it must provide verifiable proof of its real resource capacity and performance. Existing nodes then validate and vote on its admission based on a reputation-based consensus. This process significantly increases the difficulty for attackers to cheaply generate large numbers of malicious nodes. For attacks targeting the sharding mechanism, such as an adversary concentrating resources on a specific shard to disrupt its consensus, SPTSCA employs dynamic node migration and periodic re-sharding as effective defences. The system periodically reorganises shard memberships, preventing attackers from maintaining prolonged associations with particular shards. This strategy increases the difficulty of locking or corrupting specific targets. Moreover, the node migration policy incorporates historical behaviour and reputation evaluation, prioritising high-reputation nodes for migration to shards that are more critical or under heavier load. This proactive reinforcement helps strengthen potentially vulnerable parts of the system. Together with the consensus mechanism, these design elements ensure that SPTSCA maintains a robust security posture while achieving high performance.

5 Conclusions

At the economic data analysis level, a system framework was designed and implemented that deeply integrates data mining with blockchain technology. This framework effectively processes multi-source heterogeneous economic data and automatically identifies potential risk patterns through association rule mining and time-series analysis. It provides regulatory agencies with a systematic approach for accurately detecting abnormal economic behaviours within large-scale datasets. At the blockchain technology level, an innovative security-performance trade-off shard consensus algorithm (SPTSCA) was developed. By integrating dynamic sharding with an optimised reconfigurable consensus mechanism, SPTSCA significantly enhances blockchain throughput and scalability when processing high-concurrency economic transactions, while maintaining data integrity and consistency. This approach presents a new technical pathway and practical paradigm for the application of blockchain in economic data management scenarios that require both high performance and strict regulatory compliance.

Building upon the proposed architecture, future research may advance along several technical directions. One promising direction involves the integration of advanced cryptographic primitives, such as zero-knowledge proofs, into the SPTSCA framework. This integration would enable compliance verification without revealing raw transaction details, thereby achieving a balance between transparency, auditability, and privacy protection. Another direction concerns leveraging the system's low latency and high throughput to develop real-time detection algorithms and smart contract logic tailored to specific financial crime patterns, including cross-border money laundering and market manipulation behaviours. These efforts aim to translate the system's performance advantages into proactive regulatory intelligence and real-time risk interception capabilities. Together, these future directions will further enhance the practicality, adaptability, and proactive resilience of the proposed system in complex real-world regulatory environments.

Availability of data and material

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Declarations

The authors have declared that no competing interests exist.

References

Abdelhalim, A.M. (2024) 'How management accounting practices integrate with big data analytics and its impact on corporate sustainability', *Journal of Financial Reporting and Accounting*, Vol. 22, No. 2, pp.416–432.

Al-Okaily, M. and Al-Okaily, A. (2025) 'Financial data modeling: an analysis of factors influencing big data analytics-driven financial decision quality', *Journal of Modelling in Management*, Vol. 20, No. 2, pp.301–321.

Bakır, H., Ağbulut, Ü., Gürel, A.E., Yıldız, G., Güvenç, U., Soudagar, M.E.M. et al. (2022) 'Forecasting of future greenhouse gas emission trajectory for India using energy and economic indexes with various metaheuristic algorithms', *Journal of Cleaner Production*, 1 August, Vol. 360, p.131946.

Bhuiyan, M.A., Dinçer, H., Yüksel, S., Mikhaylov, A., Danish, M.S.S., Pinter, G. et al. (2022) 'Economic indicators and bioenergy supply in developed economies: QROF-DEMATEL and random forest models', *Energy Reports*, November, Vol. 8, pp.561–570.

Chukwuma-Eke, E.C., Ogunsola, O.Y. and Isibor, N.J. (2022) 'A conceptual approach to cost forecasting and financial planning in complex oil and gas projects', *International Journal of Multidisciplinary Research and Growth Evaluation*, Vol. 3, No. 1, pp.819–833.

Ghauri, S.P., Ahmed, R.R., Štreimikienė, D. and Štreimikis, J. (2020) 'Forecasting exports and imports by using autoregressive (AR) with seasonal dummies and Box-Jenkins approaches: a case of Pakistan', *Inžinerinė Ekonomika*, Vol. 31, No. 3, pp.291–301.

Gongada, T.N., Babu, B.K., Naga Ramesh, J.V.V., Syamala Rao, M.P.N., Saravanan, K.A., Swetha, K. and Tripathi, M.A. (2024) 'Enhanced quantitative financial analysis using CNN-LSTM cross-stitch hybrid networks for feature integration', *International Journal of Advanced Computer Science & Applications*, Vol. 15, No. 9, pp.755–765.

Jabeur, S.B., Mefteh-Wali, S. and Viviani, J.L. (2024) 'Forecasting gold price with the XGBoost algorithm and SHAP interaction values', *Annals of Operations Research*, Vol. 334, No. 1, pp.679–699.

Jiang, Y., Feng, T. and Huang, Y. (2024) 'Antecedent configurations toward supply chain resilience: the joint impact of supply chain integration and big data analytics capability', *Journal of Operations Management*, Vol. 70, No. 2, pp.257–284.

Kalamara, E., Turrell, A., Redl, C., Kapetanios, G. and Kapadia, S. (2022) 'Making text count: economic forecasting using newspaper text', *Journal of Applied Econometrics*, Vol. 37, No. 5, pp.896–919.

Kim, S.I. (2022) 'ARMA-GARCH model with fractional generalized hyperbolic innovations', *Financial Innovation*, Vol. 8, No. 1, p.48.

Korinek, A. (2023) 'Generative AI for economic research: use cases and implications for economists', *Journal of Economic Literature*, Vol. 61, No. 4, pp.1281–1317.

Kumar, D., Singh, R.K., Mishra, R. et al. (2024) 'Big data analytics in supply chain decarbonisation: a systematic literature review and future research directions', *International Journal of Production Research*, Vol. 62, No. 4, pp.1489–1509.

Kumbure, M.M., Lohrmann, C., Luukka, P. and Porras, J. (2022) 'Machine learning techniques and data for stock market forecasting: a literature review', *Expert Systems with Applications*, 1 July, Vol. 197, p.116659.

Lee, I. and Mangalaraj, G. (2022) 'Big data analytics in supply chain management: a systematic literature review and research directions', *Big Data and Cognitive Computing*, Vol. 6, No. 1, p.17.

Lehmann, R. (2023) 'The forecasting power of the IFO business survey', *Journal of Business Cycle Research*, Vol. 19, No. 1, pp.43–94.

Li, X., Wang, J. and Yang, C. (2023) 'Risk prediction in financial management of listed companies based on optimized BP neural network under digital economy', *Neural Computing and Applications*, Vol. 35, No. 3, pp.2045–2058.

Li, Z., Yu, H., Xu, J., Liu, J. and Mo, Y. (2023) 'Stock market analysis and prediction using LSTM: a case study on technology stocks', *Innovations in Applied Engineering and Technology*, pp.1–6.

Mehmood, K., Jabeen, F., Rashid, M. et al. (2025) 'Unraveling the transformation: the three-wave time-lagged study on big data analytics, green innovation and their impact on economic and environmental performance in manufacturing SMEs', *European Journal of Innovation Management*, Vol. 28, No. 6, pp.2189–2216.

Rao, C., Zhang, Y., Wen, J., Xiao, X. and Goh, M. (2023) 'Energy demand forecasting in China: a support vector regression-compositional data second exponential smoothing model', *Energy*, 15 January, Vol. 263, Part C, p.125955.

Sabirov, H.N. and Abduvaliyeva, Z. (2022) 'Econometric modeling of business processes based on time series data', *Экономика и Социум*, Vol. 7, No. 98, pp.102–110.

Salisu, A.A., Lasisi, L. and Tchankam, J.P. (2022) 'Historical geopolitical risk and the behaviour of stock returns in advanced economies', *The European Journal of Finance*, Vol. 28, No. 9, pp.889–906.

Tiozzo Pezzoli, L. and Tosetti, E. (2022) 'Seismonomics: listening to the heartbeat of the economy', *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, Vol. 185, pp.S288–S309.

Tulli, S.K.C. (2023) 'Enhancing marketing, sales, innovation, and financial management through machine learning', *International Journal of Modern Computing*, Vol. 6, No. 1, pp.41–52.

Visser, L., AlSkaif, T. and van Sark, W. (2022) 'Operational day-ahead solar power forecasting for aggregated PV systems with a varying spatial distribution', *Renewable Energy*, Vol. 183, pp.267–282.

Wang, J., Ma, F., Bouri, E. and Zhong, J. (2022) 'Volatility of clean energy and natural gas, uncertainty indices, and global economic conditions', *Energy Economics*, Vol. 108, p.105904.

Zheng, Y., Xu, Y. and Qiu, Z. (2023) 'Blockchain traceability adoption in agricultural supply chain coordination: an evolutionary game analysis', *Agriculture*, Vol. 13, No. 1, p.184.

Zhu, P., Hu, J., Li, X. and Zhu, Q. (2021) 'Using blockchain technology to enhance the traceability of original achievements', *IEEE Transactions on Engineering Management*, Vol. 70, No. 5, pp.1693–1707.