Journal of Systems Science and Information
Feb., 2025, Vol. 13, No. 1, pp. 137–156
DOI: 10.21078/JSSI-2024-0108

The Application of Blockchain Technology in Multimodal Educational Resources

Wenjuan CHEN
School of Humanities and Law, Wuxi Taihu University, Wuxi 214100*, China*
E-mail: Chenwj@wxu.edu.cn

Zhengjie TANG
College of International Education, Wenzhou University, Wenzhou 325035*, China*
E-mail: 371453827*@qq.com*

Jie BAI*
College of International Education, Wenzhou University, Wenzhou 325035*, China*
*E-mail: wxbaijie@*163*.com*

Abstract With the rapid advancement of information technology, the sharing of educational resources has become an integral component of the modern educational system. However, traditional data elements in educational resources face challenges such as data security, copyright protection, and trust mechanisms. Blockchain technology, as a distributed ledger technology, offers innovative solutions for the sharing of educational resources data elements with its characteristics of immutability, decentralization, and transparency. This paper designs an educational resource sharing platform based on blockchain technology, and experiments have demonstrated that by optimizing the blockchain threshold elimination model, the overall performance of the platform can be enhanced, improving the security of data processing and achieving satisfactory results.

Keywords blockchain; educational data elements; threshold elimination model; resource sharing platform

1 Introduction

The progress of information technology is causing a remarkable and unprecedented change in education. In light of this situation, the exchange of multimodal instructional resources has gained significant importance[1]. Multimodal educational resources include a wide range of information formats, including written word, visual imagery, audio recordings, and video recordings, and also involve the sharing and engagement of knowledge through many platforms

Received July 31, 2024, accepted January 7, 2025
Supported by Wenzhou University New Liberal Arts Interdisciplinary Research Project of 2023; 2023 Wenzhou University International Chinese Language Education Research Topics; Ministry of Education Industry-Academia Collaborative Education Project 2022.
*Corresponding author

and tools[2]. The distribution of these resources among various individuals helps to cater to the requirements of diverse learners and fosters the advancement of customized learning. Firstly, the act of sharing multimodal educational resources can overcome limitations related to location and time, hence decreasing the expenses associated with education. Additionally, it facilitates the creation of customized educational routes. Moreover, the act of exchanging multimodal educational resources promotes creativity in educational technology and improves the process of acquiring knowledge. In educational resource sharing, data elements can help us describe the attributes of educational resources, such as title, creator, subject, description, publisher, contributor, date, type, format, language, rights, relationships, etc. These data elements are crucial for the classification, retrieval, sharing, and utilization of educational resources.

Nevertheless, there are numerous obstacles that emerge when attempting to efficiently distribute instructional resources. One of the primary challenges in the sharing of educational resources is navigating copyright regulations and intellectual property rights. The copyright protection of educational resources limits learners' unrestricted access to high-quality knowledge.

Educational Data Elements' security and privacy concerns become of utmost importance when resources are shared online. It is imperative to safeguard the private data of pupils and educators from hackers, illegal access, or misuse. Enforcing quality standards for shared resources is crucial but can pose challenges when dealing with a wide range of institutions and individuals. Furthermore, the absence of motivating methods and the limited potential for data exchange pose significant problems in the field of educational resource sharing.

Artificial intelligence offers a variety of technologies for retrieving information from instructional materials, such as text, images, and voice. Deep learning is extensively employed in the processing of natural language for text classification[3, 4], whereas neural network algorithms are utilized for image recognition[5]. With the rise of user-generated content (UGC), educational materials now include video and live broadcast formats that involve frequent scene changes. Blockchain technology is being implemented in the sharing of educational resources, offering benefits such as intellectual property protection, privacy protection, and decentralization[6, 7].

Hence, by leveraging the technological benefits of blockchain, it is possible to construct an educational resource sharing system that is efficient, precise, and intelligent. Suggestions are proposed for the implementation of educational data elements in the stages of sharing, in-depth development, security protection, and performance evaluation.

The main progress of this study is as follows:

1) Developed a resource sharing platform based on blockchain and IPFS storage. The platform utilizes incentive mechanisms to increase users' motivation to share data resources, and utilizes the decentralized nature to ensure efficient processing of data elements. In addition, vulnerability pre-detection modules are used to audit and evaluate data content.

2) The use of DPoS (Proof of Stake) in this platform has significantly improved the incentive consensus, and implemented a multi round voting technique using threshold elimination mech-

anism to ensure that user nodes actively share data elements and contribute to maintaining economic balance on the platform.

2 Methodology

2.1 The Research Questions or Hypotheses.

This study mainly discusses how to effectively apply blockchain technology in educational resource sharing to optimize the recycling of data elements. We select multi-modal cross-platform educational resource sharing platforms and analyze how they utilize blockchain technology advantages for data element in creation, storage, retrieval, updating, and deletion.

2.2 The Data Sources and Sampling Methods.

In the pursuit of comprehensive data acquisition for this research, a multifaceted approach has been adopted, capitalizing on the rich datasets available through online educational platforms, including MOOCs, user-generated content platforms, and multimedia resource repositories. The methodologies employed for data collection are predicated on a suite of technological tools, encompassing automated data logging mechanisms inherent to the platforms, sophisticated log search analysis protocols, and web crawler data collection techniques.

The utilization of embedded data collection systems within MOOCs and analogous platforms is paramount, providing automatic documentation of learner engagement and behavioral metrics. Such metrics include the frequency of platform logins and the duration of user sessions. These captured data elements, meticulously categorized, form a foundational dataset that is readily integrated into the experimental framework of this research.

Furthermore, the application of log search analysis techniques is instrumental in chronicling and scrutinizing the events that transpire within the ecosystem of educational or learning application platforms. This encompasses a broad spectrum of data elements, ranging from learner access logs to operational records, thereby offering a granular insight into the platform's usage patterns and operational efficacy.

In addition to the aforementioned methodologies, web crawler data collection technology is employed to autonomously extract information from the web interfaces of educational platforms. This is achieved through the deployment of specialized programs or scripts, which are designed to navigate and capture data with precision. The selection of crawler frameworks is informed by their compatibility with the Hadoop ecosystem, with notable examples including Chukwa, Facebook's Scribe, Linkedin's Kafka, and Cloudera's Flume. In the data processing phase, these technologies are adept at capturing and analyzing the intricate data elements embedded within educational resources, thereby enriching the dataset and enhancing the analytical rigor of this study.

2.3 The Advantages of Blockchain Technology in the Application of Data Elements

2.3.1 Copyright Confirmation and Protection of Data Resources

Currently, a majority of countries have not yet established clear legislation regarding data copyright, leading to difficulties in managing and monetizing digital resources. In the realm of sharing educational resources, the clear delineation of copyright ownership of data is considered a crucial factor. This clear definition promotes effective management of data assets, protection of data-related rights, stabilization of data market dynamics, and the development of a strong and sustainable path for the expansion of the data sector.

In digital platforms, individuals, including corporate entities, usually hold the exclusive rights to the data they generate and the subsequent publication of information and digital works. The public datasets consist of socioeconomic development metrics, healthcare and education data, as well as information obtained via administrative law enforcement processes. The property rights of these datasets are often attributed to the governmental departments who collect or retain them.

However, there is a huge controversy surrounding the establishment of property rights for large amounts of commercial platform data. Some researchers propose that the original publishers of the data should be granted proprietary rights. Conversely, another set of researchers contends that the platform, which plays a crucial role in determining the value of the data, should be considered the legitimate owner. The difference in viewpoints creates intricacy in determining copyright, especially when many parties and data that have undergone multiple changes are involved.

Blockchain technology enables the accurate identification of owners of digital resources and the measurement of their individual contributions through the iconic elements of data elements. The cryptographic methods, such as time-stamp, hashing, cryptographic asymmetric key cryptography, and the sequential block structure, create an effective structure for determining digital property rights.

In the blockchain architecture, data is contained within blocks, which are divided into headers and bodies. The block header includes the cryptographic hash of the previous block (which is referred to as the PreHash), the hash of the transactions in the current block (Merkle root), and a timestamp, along with other metadata. On the other hand, the block body exclusively holds the transaction data that is specific to the blockchain. The blockchain's immutability is supported by the cryptographic connection between blocks, where each succeeding block includes the hash of its predecessor, resulting in a distributed ledger that is resistant to tampering. When users want to publish or copyright digital goods, the data undergoes cryptographic hashing, timestamp, and encapsulation within a block. The process of registering digital assets on the blockchain involves timestamp and authentication using the user's private key. This ensures the verification of the user's identity and the exact moment when the asset was registered or created on the blockchain.

The integrity of the blockchain is reinforced by the cryptographic hashing of blocks, making it practically impossible to modify the registration information of a block. Any attempt to do so

would require recalculating the hash of all future blocks. The blockchain possesses the attribute of immutability due to its inherent trait.

Furthermore, smart contracts function as digital agreements that regulate the acquisition and transfer of digital property rights, thereby establishing a contractual structure for transactions and safeguarding rights inside the blockchain ecosystem[8]. Together, these characteristics make blockchain technology a powerful tool for securely and transparently managing digital resource transactions and protecting against infringement.

The decentralization, immutability, and high transparency of blockchain can provide technical guarantees for high efficiency and security in the circulation of data elements. The specific working process can be seen in Figure 1.

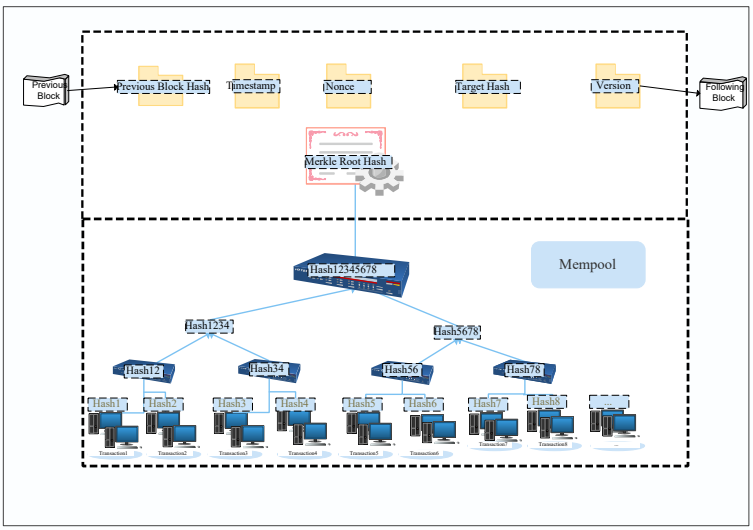

Figure 1 Data block structure diagra

2.3.2 Consortium Blockchains Offer Superior Privacy Protection for Data Elements.

Within the realm of blockchain technology, the structure of public blockchains represents a model of complete decentralization and unfettered entry. These networks function via an open invitation protocol, allowing for widespread participation, supervision, and examination of transactions. This promotes a system that is fair and transparent. Nevertheless, the characteristic of being open might potentially undermine privacy, which is a significant worry that goes against the security requirements necessary for the shared use of digital assets.

Regarding these limitations, the suggestion of consortium blockchains emerges as a more feasible option. These blockchains have a semi-decentralized structure and are managed by a consortium of entities. Each entity must go through a vetting process and be accepted by the collective membership. Blockchains can be operated either publicly or in a controlled manner, with a strong emphasis on protecting privacy. Consortium blockchains are very useful for creating shared copyright management systems including various stakeholders. This enables a

cooperative method for verifying the authenticity of copyright, reducing piracy, and ensuring a fair sharing of profits.

This paradigm transition from public to consortium blockchains emphasizes the potential to improve the integrity and confidentiality of digital resource management, while also addressing the communal objectives of economic equity and copyright protection. Consequently, the consortium model is established as a strategic framework for the management of digital intellectual property within a secure, yet cooperative ecosystem.

It is important to note that the China Copyright Protection Center, in partnership with Weibo, has jointly established the DCI Standard Consortium Blockchain as shown in Figure 2. This blockchain can offer comprehensive system support for the shared copyright protection of digital resources.

Blockchain can support classification and hierarchical management of data based on different keyword elements, set different security policies and access control policies according to different security levels, achieve automatic verification through smart contracts, and use distributed consensus mechanisms to achieve cross platform sharing of data circulation. This not only provides effective basis for cross subject data identification and verification, but also guarantees the privacy of individual data elements of data owners.

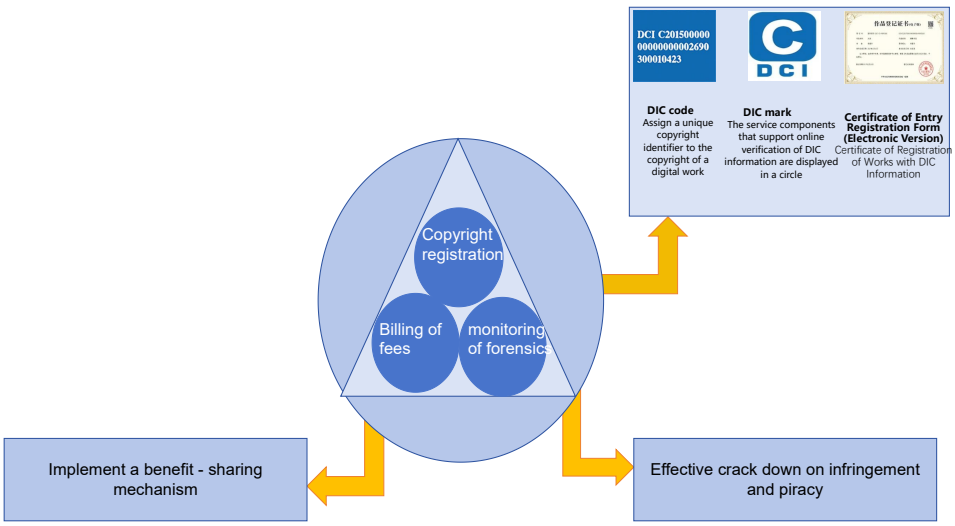

Figure 2 Consortium blockchain framework diagram

2.3.3 Smart Contracts Play a Protective Role in Digital Asset Infringement

Currently, the infringement of digital resources is characterized by its clandestine nature, cross-jurisdictional propensity, and the rapid dissemination of illicit activities. Within this context, the incorporation of smart contracts, which possess an inherent capacity for autonomous execution, offers a paradigm-shifting solution. The requisite restrictive or remedial actions are initiated immediately upon the detection of an infringement that corresponds with the stipulated parameters incorporated within a smart contract, thereby avoiding the delays that are typically

associated with traditional legal recourse.

In addition, smart contracts have the potential to record important actions and transactional information in real-time, creating a comprehensive and unchangeable record of occurrences. This ledger functions as an uncorruptible chain of evidence, greatly enhancing the dependability and openness of the data crucial for determining culpability in situations of infringement.

2.3.4 Incentive Mechanisms Play a Positive Role in the Sharing of Digital Elements

The implementation of blockchain-based incentive protocols fosters the emergence of a beneficial data circulation paradigm. Within this framework, contributors of educational content are remunerated with digital incentives, which, in turn, augment their motivation to engage in further contributions to the system.

In the initial phase, the blockchain framework employs incentive mechanisms, exemplified by token-based rewards, as a strategic approach to galvanize an expanded cohort of creators and publishers to participate in the network dedicated to the communal sharing of resources. This engagement is perceived as a catalyst for bolstering the collaborative ethos of the digital community.

Subsequently, it is believed that the incentivization protocols included in blockchain technology will have a beneficial impact on the quality of digital resources. Token awards are strategically allocated across the blockchain platform ecosystem to motivate content providers to produce high-quality and substantial content. This technique is crucial for improving the overall quality of material and providing readers with a more reliable and excellent source of information.

Furthermore, the blockchain's incentive systems, which reward and compensate users that showcase innovation within the network, are seen as a catalyst for the rapid advancement of technology and the widespread adoption of various applications. This dynamic promotes a creative atmosphere and enhances the overall growth and resilience of the digital ecosystem.

2.4 The Main Differences Between This Study and Traditional Sharing Systems

Compared to previously proposed educational resource sharing systems, blockchain-based educational resource sharing platforms enhance the robustness and security of the system through their distributed storage architecture. The immutability of blockchain ensures the protection of copyright and the quality of educational content. The automatic execution mechanism of smart contracts improves the efficiency of resource sharing. Furthermore, the platform's incentive mechanisms, evidentiary capabilities, and performance optimization features significantly enhance the security, transparency, and efficiency of educational resources.

3 System Design and Experimental Deployment

3.1 Blockchain Environment Deployment

The test system of the model operates on the Ethereum Sepolia chain and utilizes the geth client to establish the blockchain environment[9]. Optimal setup parameters: CPU must be

either Intel i5 or AMD Ryzen 5 or a higher model. Minimum requirement: 8 GB or more RAM. The reason for using Disk over Sepolia as a test network is its modern infrastructure, which is more conducive to handling data. It is recommended to use geth version 1.13.11 or later for better control and compatibility.

Since the inherent potential of digital educational resources to work on several platforms, platforms requires them to be compatible and easily transferable, it is crucial to use a Node.js server for distribution. This server enables the smooth integration and transfer of instructional information between different operating systems, hence improving accessibility and usefulness.

Furthermore, the implementation of the inter planetary file system (IPFS) is crucial in this context. IPFS functions as a decentralized protocol that facilitates the development of a resilient and enduring distributed storage solution. It enables the exchange of files via a network transmission technique that is intrinsically optimized for efficiency and scalability. This technique efficiently addresses the difficulties related to storing large amounts of data, while also decreasing the financial impact of data storage on the main blockchain infrastructure.

Integrating IPFS into a Node.js server-based system improves the administration of digital educational resources and enhances the sustainability and cost-effectiveness of data management in blockchain-based educational platforms.

3.1.1 Install MongoDB

Installing and configuring MongoDB on a Linux server involves performing a sequence of instructions. This procedure involves the retrieval and transfer of the installation package, configuration of environmental variables, creation of data and log pathways, and activation of the MongoDB service. MongoDB's parameters may be conveniently customized with a configuration file, which contains settings for the database location, log path, port, and other pertinent information, making maintenance easier.

3.1.2 Install Node.js

The installation of Node.js is usually done by downloading the source code compilation or using a package manager such as apt or yum. Specific versions of Node.js can also be installed through the Node Source RPM repository. After the installation is completed, the node-version command can be used to verify the success of the installation. Furthermore, the configuration of npm mirrors can accelerate the subsequent package installation process.

3.1.3 Configure Express Application

In order to start, it is necessary to globally install Express and a project generator. Afterward, utilize the Express project Generator to establish a new web application project. Install the required dependencies in the project and set up routers, controllers, views, and other components. Express applications have the capability to handle HTTP requests via several routes and send back responses to the client.

3.1.4 Connecting to MongoDB with Node.js

The interaction between a Node.js application and MongoDB can be facilitated through the mongoose module. The initial step involves installing the mongoose module, followed by scripting to define data models and execute database operations.

To ensure that the application automatically establishes a connection to MongoDB upon startup, it is necessary to edit the pertinent JavaScript files to configure the database connection details. This configuration guarantees that the application is equipped to seamlessly interact with the MongoDB database as part of its initialization process.

3.1.5 Deploy and Launch the Web Application

Process management tools such as PM2 can effectively manage and monitor Node.js applications. Begin by globally installing PM2, and then use PM2 to launch the application from the project's root directory.

To ensure that the deployed application is functioning correctly, access it using the server's public IP address combined with the port number. If access via the public internet is required, it may also be necessary to configure firewall and security group rules.

3.1.6 In the App Directory, Run and Start the Nodemon Service

These are all the procedures required to launch this project. After the deployment is finished, initiate the local MetaMask wallet. Then, within the test account, one have the ability to utilize your own test wallet ETH to carry out transactions involving data resources.

3.2 Design and Operation of a Blockchain-based Educational Resource Sharing Platform

3.2.1 Platform Overall Design and Main Modules

To construct a high-quality content database with timely updates and a large quantity of smart contracts, this study has improved the delegated proof of stake (DPoS) incentive algorithm and verified the feasibility of the proposed solution. The overall design of the platform and its main modules is shown in the following figure.

The scheme design diagram illustrates that user nodes upload digital resources, which then trigger the automated execution of smart contracts. Once the resources have been scanned for vulnerabilities and processed by the middleware, they are then transferred to IPFS. After confirming the lack of duplicates, IPFS transfers the resources to the database of the processing middleware. Afterwards, the processing middleware records a distinct content identification onto the Ethereum blockchain. Simultaneously, user nodes have the ability to participate in interactions with several nodes through the incentive mechanism. These interactions can involve activities such as creating content, trading, and voting.

The incentive mechanism module has been constructed via a multi-node approach, classifying user nodes into three specific types: Ordinary nodes, verification nodes, and accounting nodes. This enhances the delegated proof of stake (DPoS) protocol and incorporates aspects from

```
// Connecting to MongoDB with Node.js

// Import the 'mongoose' module to interact with
MongoDB
const mongoose = require('mongoose');

// Define the data model using mongoose's model
schema
const DataModel = mongoose.model('DataModelName',
new mongoose.Schema({
  // Define schema fields here
}));

// Function to handle database operations
async function performDatabaseOperations() {
  try {
    // Connect to MongoDB using mongoose
    await
mongoose.connect('mongodb://localhost:27017/yourD
atabase', {
      useNewUrlParser: true,
      useUnifiedTopology: true,
    });

    // Perform database operations using the
defined model
    // e.g., const result = await
DataModel.find({});
    console.log('Connected to MongoDB and
performed operations.');
  } catch (error) {
    console.error('Error connecting to MongoDB:',
error);
  }
}

// Ensure the application connects to MongoDB on
startup
performDatabaseOperations();

// Example of a route that uses the database
model to respond to client requests
const express = require('express');
const app = express();

app.get('/data', async (req, res) => {
  try {
    const data = await DataModel.find({});
    res.json(data);
  } catch (error) {
    res.status(500).send('Server error');
  }
});

// Start the Express server
const PORT = process.env.PORT || 3000;
app.listen(PORT, () => {
  console.log(`Server running on port ${PORT}`);
});
```

Figure 3 Deploy the compilation interface

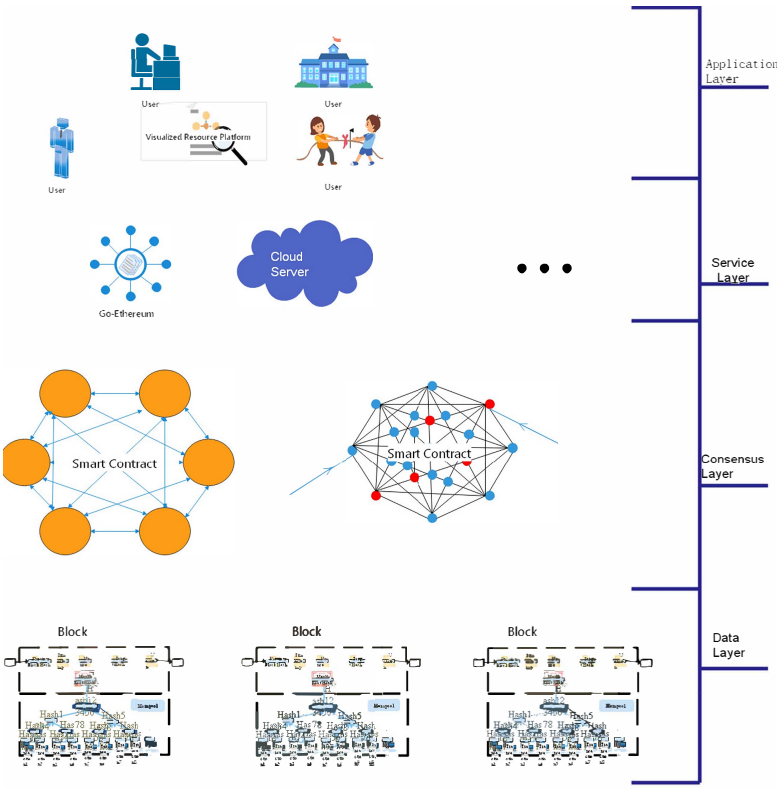

Figure 4 System design diagram

WEB3 projects. This module is specifically created to improve user involvement by utilizing smart contract data and ensuring a clear allocation of tasks among different types of nodes. Ordinary nodes have the responsibility of uploading digital resources that they have copyright for. Verification nodes are in charge of auditing and validating information, identifying weaknesses, and categorizing them. Accounting nodes supervise the creation of blocks and the recording of transactions within the blockchain system. Nodes can receive community contribution points or token incentives upon successfully executing their obligations. These benefits can be exchanged for smart contracts inside the platform[10].

Verification and accounting nodes are periodically elected by ordinary nodes through a voting process. A multi-round election verification approach has been implemented to guarantee authenticity. This method is based on a threshold elimination mechanism. After each voting round, nodes that have a prestige value lower than the specified threshold are penalized by decreasing their voting weight. This penalty applies to verification and accounting nodes, in addition to the ordinary nodes that voted for them. If any verification or accounting node behaves improperly during its time in office, its reputation and voting privileges will be reduced based on user reports and administrator verification.

This mechanism also ensures a fair distribution of rights and responsibilities among each node, so limiting the introduction of harmful information that may result in data corruption or

intentional sabotage. As a result, the entire platform operates smoothly and without disruption.

In order to evaluate the information and content of smart contracts, this system implements dynamic vulnerability detection tools, including ArtemisX and SmartCheck[11]. If a contract successfully passes the detection, it will be deposited in a pending pool, where it will await manual review and vulnerability level annotation by the verification nodes. The verification nodes will conduct an objective overall assessment of the contract from multiple perspectives, including the authenticity of the information, code legibility, and functional completeness, in accordance with the grading strategy of the scheme, and designate it the corresponding level. The contract will be published to the InterPlanetary File System (IPFS) by the ordinary node that processed the contract upload after the verification nodes have completed the review and annotation duties. Subsequently, the IPFS system will undergo activities such as duplicate detection[12].

3.2.2 Architecture of Smart Contract Sharing Platform Based on Blockchain

The architecture of the platform mainly includes four layers: Application layer, service layer, consensus layer, and data layer[13]. The application layer comprises two components: The user interface and the user. Its primary function is to offer services such as account administration, contract management, and incentive management. The interface serves as the medium via which users engage with the platform. The user interface serves as a starting point for user actions, with the user being the designated operator. The application layer is accountable for accepting user requests, transmitting them to the subsequent layer for processing, and delivering the processing outcomes back to the user.

The service layer primarily comprises two components: Cloud servers and Go Ethereum. The cloud server offers a reliable service environment and resource support, referred to as the backend, for deploying different adaption environments of the platform and achieving diverse purposes. Go Ethereum offers the fundamental blockchain protocol implementation for the Ethereum private chain. It packages and delivers this implementation to servers or users in the form of an API or command-line interface for use.

The consensus layer primarily utilizes optimized incentive consensus contracts and Solidity-based smart contracts to implement functional control[14]. These contracts are developed in a multi-node mode and are accompanied by incentive mechanisms. The consensus layer serves as the fundamental framework for implementing a wide range of business functions.

The data layer consists of three components: MySQL, IPFS, and Ethereum blockchain, which are utilized to facilitate the service layer. Among these components, MySQL is connected to the processing platform and is tasked with storing user information, smart contract information, and other data. IPFS offers users efficient and dependable file storage services. Blockchain is responsible for storing the content identifiers and other fundamental information of smart contracts published by IPFS, while also ensuring secure and reliable data protection.

This research has established several sub-modules, including an incentive module, pre-detection module, user management module, data management module, and contract processing

module, for the platform. These sub-modules have been constructed in accordance with the functional needs of the platform, following a framework of low coupling and modularization[15].

The platform's information verification and auditing smart contract is solely activated by administrators and is specifically designed to automate the auditing of user identification and node information[16]. The system utilizes an algorithm similar to Know Your Customer (KYC) procedures to automatically assess the compliance of user-submitted information. This includes verifying the authenticity of identifying papers and evaluating the frequency of information changes. After successful verification, the processing middleware and the administrator nodes agree to provide operational rights, particularly in conjunction with the features of the user management module.

The incentive consensus module is an essential element in guaranteeing the long-term functioning of the platform. Initially, it utilizes the Proof of Work with Public Data Set (PoWPDS) approach to guarantee ongoing user engagement and involvement. In addition, it engages with other submodules, such as the user management module, to ensure the platform's usefulness and integrity.

Preliminary Detection Module: The preliminary detection module is integrated with automated detection and manual verification procedures. The automated vulnerability detectors are used to analyze the node information and smart contracts that are uploaded. Contracts that are not compliant are returned for correction. The automated detection process flags contracts that pass and directs them to a manual review. Verification nodes refine the tags and categorize them accordingly.

Following a successful initial detection, the contracts and node information are uploaded to the InterPlanetary file system (IPFS), marking the completion of the initial detection phase. This technique guarantees that only contracts of superior quality are accepted into the platform, which not only increases the platform's attractiveness but also providing incentives for node users. The strict and proactive screening process is crucial for preserving the integrity and dependability of the platform's ecosystem.

The user administration module includes features for verifying user identification during registration and managing user information. Upon registration, new node users must provide their identification information, which will undergo review by the administrators to finalize the verification procedure. When users join, they establish synchronization with the blockchain and IPFS networks. The information is kept in MySQL databases and backed up as necessary. This approach serves to thwart the infiltration of malevolent nodes, thus bolstering the security of the platform.

Once a node has joined, it has the right to update its information. Nodes selected via voting for verification or accounting duties must synchronize their new information and permissions. The node triggers the process of updating information, and then the platform sends a request to update permissions. Once the administrators have reviewed the revised information, it is synced with both the blockchain and IPFS. At the same time, the MySQL database is updated,

marking the completion of the procedure.

The data management module encompasses functionalities for data storage, retrieval, and dissemination. It utilizes blockchain, IPFS, and MySQL for storage, adhering to the notion of three paradigms. During the process of storing data, IPFS functions as a node in a private network that saves information and contracts. Blockchain is responsible for recording content identifiers and transaction information. MySQL is used to store user information, pre-detection results, and server logs.

During the data query and content distribution process, users get contract data by using keywords or scopes. The processing center then matches and retrieves the corresponding results from the database. When users engage in contract exchanges, the processing center facilitates the completion of token transactions. Once the transaction is completed, the intermediary platform sends the content identifier to IPFS and grants the user access to the contract. Users synchronize the content of the contract in IPFS and process the recorded transaction information though the middleware in MySQL.

Contract Processing Module: A multi-round election and voting verification method based on a threshold elimination mechanism is developed to select higher-level nodes from ordinary nodes. This method aims to ensure the validity of voting and reduce malicious activities orchestrated by interest groups consisting of stakeholders. It extends the principles of delegated proof of stake (DPoS) voting. After each voting round, nodes who do verification and bookkeeping and ordinary nodes that voted for them, will be penalized and their voting weights will be decreased if their Prestige (P) values are below a specified level. Furthermore, if verification or bookkeeping nodes engage in any wrongdoing throughout their period of service, users can report it and, following verification by administrators, deductions will be made from their Prestige and voting rights. This strategy enables the effective distribution of jobs across different nodes, allowing each node to make the most of its resources while reducing the chance of failures and promoting collaboration and specialized roles among nodes.

3.2.3 Mathematical Model for Optimizing Blockchain Threshold Elimination Mechanisms

When optimizing the mathematical formula for a blockchain threshold elimination mechanism, it is important to take into account crucial elements such as the quantity of votes collected by candidates, the voting power of community members, and the dynamics of the elimination process. The optimization model is presented in the following manner:

Basic Assumptions:

N: Total number of candidates

V_i: The number of votes received by the i candidate

W_j: The weight of the j voter

T: Thresholds set

R: Number of representatives elected per round

K: Total number of voters

Dynamic threshold model

$$T = f(N, R, K), \tag{1}$$

f is a function that dynamically adjusts threshold values based on the total number of candidates N, the number of elected representatives R, and the total of voters K.

Weight adjustment model

$$W_j' = W_j \times (1 + \alpha + P_j), \tag{2}$$

W_j': adjusted weight

P_j: the historical performance score of j

α: the adjustment coefficient used to adjust the weight according to the historical performance

Voting Weighted Model

$$V_i' = \sum_{j=1}^{K} (W_j' + v_{ij}), \tag{3}$$

V_i': weighted number of votes for the i candidate

v_{ij}: is the vote from the j voter for the i candidate (0 or 1)

Threshold Elimination Mechanism Model

After each round of voting, the weighted number of votes for each candidate is calculated V_i', then sort according to V_i', and eliminate the candidates whose number of votes is below the threshold T.

Multi-round elimination model

If the initial round does not have an adequate number of representatives chosen, a multi-round elimination process may be used, and the threshold T for each round can be altered dynamically:

$$T_{\text{round},k} = T_{\text{initial}} \times (1 - \beta \times (k - 1)), \tag{4}$$

$T_{\text{round},k}$: the threshold for the kth round

T_{initial}: initial threshold

β: the percentage decrease per round

Motivation and punishment Candidates that receive a significant number of votes or have a strong track record might be incentivized by raising their influence in the following round. Conversely, candidates who perform poorly can be penalized by reducing their influence or eliminating them outright.

Improved nodal voting patterns are as follows:

In the electoral strategy based on the threshold elimination mechanism, new nodes are assigned the initial Reputation Value, and daily behavior affects the reputation value, resetting voting rights to 1 per election period. At the end of each round of voting, the voting rights of

nodes with a reputation below the lower threshold are abolished and the voter weight of those nodes supported is reduced. Instead, voters who supported highly reputable nodes increase their weight by Y. This mechanism encourages honest behavior and curbs malicious attacks.

Implementation of incentive methods: The modular approach to building contracts reduces complexity and improves code readability. Interaction between modules is easier to manage. Node functions in the system are broken down into minimum operations, i.e. transfers between different nodes, including upload, audit, block-out, dissemination and exchange transactions. These operations are carried out by processing calls from the user-end blockchain. For example, after users successfully upload and validate smart contracts, the main node is transferred to the user, and funds are synchronized in the blockchain network and platform databases. The nodeInfo object that is returned includes many elements, such as enode (the URL of the node's enode), id (the ID of the node), name (the name and version of the client), and protocols (a list of the protocols that are supported). Next, incorporate the primary node using the admin.add Peer command, which facilitates the automated synchronization of block contents among all nodes once the connection is established.

Once the connection between multiple nodes is established, the block information can be queried and synchronized. The main node account can be unlocked, and transactions can be made between the main nodes and specified nodes by sending a transaction command. The transfer order is then issued, and the account node mines, completing the transactions and recording them in the block. Finally, the balance of the two accounts can be queried to confirm the transfer amount of 1 token, with each account showing a balance fluctuation of $1 * 10^{19}$ Wei.

Once the minimum executable function authentication is completed, including tasks such as establishing multi-node connectivity, synchronizing, and transfering, it is essential to utilize the contractual and processing points that were initially connected to these functions. In order to enable the multi-node mode on the platform, the PoWPDS mechanism is used for income distribution. This is accomplished by implementing assignment control through the creation of smart contracts. The smart contract utilizes the need function in the allocation process to verify if the transfer meets certain conditions, such as having adequate balance in the transfer node and a valid recipient address. Once the transfer is finished, the emit keyword activates a Transfer event to document the specifics of the transaction. The key code is as stated.

4 System Testing and Experimental Validation Results

As blockchain technology remains in its infancy, there are no national test standards available. Thus, this paper follows the commonly utilized China blockchain technology and industry development forum standard to conduct a comprehensive test of the program, involving functional testing and performance testing. Results of the functional module test are presented in Table 1.

Functional testing has demonstrated that the system's blockchain functionality is capable of

Table 1 Test results of functional modules

Serial NO.	Testing category	Test items	Test sub item	Result
1	User Function	user interface	Command line interaction	T
2			Graphic interaction	T
3			Gapplication program Interface interaction	T
4		Transaction submission	Transaction submission function	T
5		financial management	financial management function	T
6		Member Management	Identity management	T
7			Permission management	T
8			Data confidentiality	T
9			Audi table function	T
10		Business Function	Fault detection	T
11		Monitoring management	Network operation	T
12			Status monitoring	T
13		Problem management	Network issues and Tracking and Reporting	T
14		Security management	Account security function and Tracking and Reporting	T
15	Access management	Account information inquiry	Basic Account Related information	T
16		Account book information query	Total block height Degree inquiry	T
17			Block identification query	T
18		Transaction operation processing	Specific transaction operations Submit a request	T
19		Interface service Ability management	Interface call Frequency management	T
20			Interface query Cache processing	T
21		Interface access Permission management	Lower level Permission interface	T
22			Higher level Permission interface	T
23		Node Server	Node Server Information Service	T
24		Node Service Control	Node startup	T
25			Node shutdown	T
26			Node participation Identify	T
27			algorithm configuration	T
28		Node service configuration	Node Connection Quantity configuration	T
29			Node provides external services Access service configuration	T
30		Node authorization management	admittance	T
31			Accurate release	T
32		Tested node transaction processing		T
33			Specific before consensus Identifying assets Logical verification	T

satisfying business requirements and reliably functioning when the number of participants and test duration are raised. The use of multiple nodes for identification and the recording of independent information in the consensus mechanism highlights the distinct benefits of blockchain technology and effectively showcase its anti-corruption features. Additionally, the testing of smart contracts to support the upgrading of multi-party contract content has proven to be reliable and consistent. Participants submitted films from a range of 2 to 20 platforms. In the future, the number of platforms will be gradually increased during follow-up to ensure continual stability testing. This verification confirms the general stability, adaptability, and scalability of video resources in the processing of cross-platform mode. The system performance test results are presented in Table 2.

Table 2 Performance test results

Serial number	Test items	Test sub item	Test Result
1	Basic account transfer Account performance testing	Basic account single transfer test	T
2		Basic account load transfer test	T
3	Smart Contract Conversion Account performance testing	Smart Contract Single Transfer Test	T
4		Smart Contract Load Transfer Test	T
5	Transaction verification test	Single account verification test	T
6		Load account verification test	T
7	Block verification	Single block verification test	
8		Load verification test	T
9	Stability testing	system stability	T

The phase test results are determined in accordance with the user rules, as illustrated in Table 3, to ascertain the stability of the system's operation over the specified period of time in the context of a progressive increase in the number of participants. The test time is set to 7 days, as the duration of the newly released video ranges from 0 to 7 days, from attracting audience interest to progressively exiting hot search. The test results indicate that the system is effectively configured to meet the specifications.

Table 3 Stage test results

Number of participants	Test time/d	test result
21	0.5	stable
222	1	stable
997	1	stable
1997	2	stable
2896	2.5	stable

The experimental design involves simulating the conditions of educational resources and

conducting a 7-day cycle experiment. In this experiment, 1567 virtual users will be randomly assigned to upload video files ranging from 30∼60 MB in size. The objective is to compare the performance of three important modules in the system platform: The incentive module, pre-detection module, and contract processing module. Specifically, the comparison will focus on the resource sharing processing speed, security, and stability of these modules.

5 Conclusion and Future Prospects

We propose a novel approach that integrates blockchain technology with multimodal digital resources to enhance the feasibility of sharing. This paper improves the threshold-based voting mechanism to increase the enthusiasm of publishers for contributing resources. The research utilizes the IPFS system, which greatly enhances the storage capacity of multi-modal cross-platform resources. The effectiveness of this method has been demonstrated in the experimental system, highlighting the potential of such an integrated strategy in the sharing of multi-platform resources.

Looking ahead, the relentless advancement in the sophistication of smart contract design and execution through technological innovation stands as a critical conduit for amplifying their efficacy in the governance of liability pertaining to infringement. The iterative enhancement of these mechanisms is projected to be instrumental in forging a more robust framework for the management of intellectual property rights in the digital sphere.

The potential issues that may arise in the future implementation of the blockchain-based educational resource sharing platform proposed in this study include: The compatibility between existing resource systems and blockchain technology, necessitating upgrades and integration; blockchain is secure, improper smart contract coding and private key management could pose risks; the lack of standards due to the nascent stage of blockchain application in the educational sector affects platform interoperability; smart contracts may not cover all resource sharing scenarios; data storage is limited; and insufficient user participation could affect the incentive mechanism, thereby impacting network effects.

References

[1] Guo J, Wu Y H, Gu L, et al. Current status and prospects of digital resources in international Chinese language education. International Chinese Language Teaching Research, 2021(4): 86–96.

[2] Xu H. Ecological analysis of the optimal allocation of digital resources in international Chinese education. Proceedings of the 2023 Higher Education Research Forum, Guilin Branch, 2023: 3.

[3] Hany F A, Muhammad A A, Ahmed G A, et al. A review of blockchain in internet of things and AI. Big Data and Cognitive Computing, 2020, 4(4): 28.

[4] Liu Y H, Chen K. New developments in blockchain consensus mechanisms. Computer Applications and Research, 2020, 37(2): 6–11.

[5] Cao M R. Research and practice on privacy data protection technology for the internet of things based on blockchain. Cyber Security, 2024, 15(1): 118–123.

[6] Zhao W J. Blockchain technology: Development and prospects. National Science Review, 2019, 6(2): 193–197.

[7] Zheng Z B, Xie S A, Dai H N, et al. Blockchain challenges and opportunities: A survey. International Journal of Web and Grid Services, 2018, 14(4): 352–375.

[8] Feist J, Grieco G, Groce A. Slither: A static analysis framework for smart contracts. 2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB). IEEE, 2019: 8–15.

[9] Mueller B. Smashing ethereum smart contracts for fun and real profit. HITB SECCONF Amsterdam, 2018, 9: 54.

[10] Mossberg M, Manzano F, Hennenfent E, et al. Manticore: A user-friendly symbolic execution framework for binaries and smart contracts. 2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE). IEEE, 2019: 1186–1189.

[11] Zhang P, Xiao F, Luo X. A framework and dataset for bugs in ethereum smart contracts. 2020 IEEE International Conference on Software Maintenance and Evolution (ICSME). IEEE, 2020: 139–150.

[12] Pierro G A, Tonelli R, Marchesi M. An organized repository of ethereum smart contracts' source codes and metrics. Future Internet, 2020, 12(11): 197.

[13] Elrom E. NEO blockchain and smart contracts. The Blockchain Developer. https://doi.org/10.1007/978-1-4842-4847-8_7.

[14] Ni Y D, Zhang C, Yin T T. A survey on smart contract security vulnerabilities. Bulletin of Information Security, 2020, 5(3): 78–99.

[15] Daniel E, Tschorsch F. IPFS and friends: A qualitative comparison of next generation peer-to-peer data networks. IEEE Communications Surveys and Tutorials, 2022, 24(1): 31–52.

[16] Albert E, Gordillo P, Livshits B, et al. Ethir: A framework for high-level analysis of ethereum bytecode. Automated Technology for Verification and Analysis: 16th International Symposium, ATVA 2018, Los Angeles, CA, USA, October 7–10, 2018, Proceedings. Cham: Springer International Publishing, 2018: 513.