

A Secure and Accessible Framework for Academic Information Using Blockchain

Parveen Kumar¹, Sanjeev Kumar Rana¹

¹Department of Computer Science and Engineering

MM Engineering College, Maharishi Markandeshwar

(Deemed To Be University)

Mullana, Ambala, Haryana-133203, India

Email: parveentanwar2006@gmail.com

Abstract

The explosive expansion of online higher education networks has caused legitimate concerns about privacy, integrity, and verifiability of educational records. Legacy systems tend to be centralized and thus typically poor in fine-grained access control and susceptible to unauthorized alterations. Addressing these, an authentication and authorization system based on secure and scalable cryptographic and blockchain technologies is introduced in this research. Synchronizing Dynamic Huffman Coding (SDHC) is employed as a preprocessing technique for redundancy reduction and encryption optimization. The preprocessed data is then encrypted with Hierarchical Attribute-Based Encryption (HABE), whereby only those users possessing the appropriate role-based and hierarchical access permission can decrypt the reports. Once encrypted, each record is hashed using the BLAKE3 algorithm, which is very performant and very collision-resistant. The hashed records are structured into (VrT) that yields smaller proof sizes and increased verifiability speed. The Verkle root is safely deposited on a blockchain via the Clique Proof-of-Authority (C-PoA) consensus protocol, which facilitates power-saving and lightweight consensus among authorized validators. The whole system is coded in MATLAB, and its performance is compared with current HSBSCS, BSDMF, and ECDSA-SHA-256 models. Experimental results show that the proposed model reduces encryption time by 26.4% and increases avalanche effect consistency by 19.7%, proving its efficiency in safe academic record management.

Keywords: *Academic data security, Hierarchical Attribute-Based Encryption (HABE), Verkle Tree, Synchronizing Dynamic Huffman Coding, BLAKE3 hash, educational records, Tamper-proof verification.*

1. Introduction

The digitalization of the education industry has facilitated an unprecedented expansion in the generation and use of education data [1]. Virtual learning spaces, online assessment technologies, virtual classes, and learning management systems currently dominate the delivery and administration of education [2]. Consequently, enormous amounts of sensitive information like student identities, performance indicators, academic history, and behavioral data are gathered and stored electronically [3]. While this information benefits personalized learning and institutional analytics, it also brings considerable security, privacy, and

accessibility risks with it [4]. Data in most educational environments tends to be stored in centralized systems owned and operated by specific institutions or third-party providers [5]. These systems do not have proper security protections and are susceptible to single points of failure [6]. Unauthorised access, hacking events, data manipulation, and system failures are now common issues, compromising the confidentiality and integrity of educational data [7]. Secondly, students and teachers often have little or no control over their data, with limited transparency about how it is accessed, moved, or used by other individuals [8]. The second is interoperability and standardization of learning data across institutions and platforms [9]. When students transfer from one level of education to another or change institutions, their learning transcripts should be securely and verifiably transferred [10].

But current systems falter in data portability, consistency, and trust. Manual verification processes are labor-intensive and susceptible to fraud, resulting in inefficiencies in admissions, credential evaluation, and employment verification background checks [11][12]. Blockchain technology offers a solution to these failings in the guise of a decentralized, transparent, and tamper-evident data management system [13]. Through its distributed ledger capability, blockchain is able to store and verify educational documents securely without relying on a central authority [14]. Smart contracts are able to access information automatically such that students are able to be in control of their credentials for studies and grant selective access to institutions or employers [15]. This decentralized system also facilitates cross-institutional collaboration and instant verification, thus making learning services more efficient and trustworthy [16]. Blockchain also promotes the establishment of safe digital identities and verifiable credentials, enabling lifelong learning and global recognition of academic achievements [17]. Students could accumulate micro-credentials, certificates, and degrees from several providers and integrate them into one unified, tamper-evident digital portfolio [18]. Not only does this empower students with data control but also streamlines administrative processes within the education ecosystem [19]. Blockchain in education, thus, can enhance data security, transparency, and greater autonomy for the students [20].

Despite immense advances in distance learning, current systems of educational data management remain broken, insecure, and opaque. Centralized designs are susceptible to hacks, are non-interoperable, and have a tendency to disempower student control over educational data. All of which complicate learning data sharing, credential verification, and global learner mobility. To address these impediments, there is a need for an urgent, secure, decentralized, and accessible framework of educational data that takes advantage of blockchain technology's immutability, transparency, and distributed nature to secure data integrity, give users greater control, and accommodate the future of digital education. As a result of these limitations, there is a need for a decentralized, secure, and interoperable framework of educational data. The research described here is necessitated by the need to protect academic records from tampering while providing students with ownership and control over their information. This research advocates for a blockchain-based solution that integrates advanced encryption, reliable hashing of data, and consensus to create tamper-proof storage, verifiable exchange of credentials, and trusted access control. The system overcomes existing weaknesses

by utilizing distributed ledger technology to increase transparency, scalability, and trust in academic record management.

The major contributions of the research are as follows,

- Introduced SDHC for effective preprocessing and minimized data redundancy before encryption.
- Utilized HABE to support fine-grained role-based access to secure academic records.
- Applied BLAKE3 hashing to facilitate quick, secure, and collision-resistant fingerprinting of encrypted data.
- Constructed Verkle Trees for scalable and lightweight proof creation, minimizing verification overhead compared to regular Merkle Trees.
- Deployed C-PoA consensus to realize energy-efficient and high-throughput blockchain verification.
- Developed a blockchain-based student data system that provides tamper-proof storage and shareable verifiable credentials.

The rest of the research is organized as follows: Section 2 presents the current literature on secure and decentralized academic data management. Section 3 outlines the proposed framework. Section 4 illustrates the experimental setup using MATLAB, performance analysis, and comparison with other approaches, and Section 5 concludes the research and offers future work directions for enhancing blockchain-based educational systems.

2. Recent Research Work: A Brief Review

This area emphasizes recent developments in secure academic record keeping by combining state-of-the-art cryptographic methods and blockchain.

In 2024, Alsulami, B.M., *et al.*, [21] have presented a Highly Secure blockchain-based Compressive Sensing (HSBCS) system, wherein blockchain technology was employed for verifying data integrity and secure biometric authentication. It was targeted toward the education sector with thorough testing and performance measurements. The results demonstrated the educational usability of the HSBCS system to students by significantly enhancing the security and legibility of personal academic records. However, the system was limited by interoperability and scalability when it was deployed in various schools with varying data standards and infrastructure capacities.

In 2024, Abbas, A., *et al.*, [22] developed a blockchain-based secure data management framework (BSDMF) for managing medical data in an Internet of Medical Things (IoMT) environment with the perspective of securely exchanging patient data while enhancing scalability and data availability within healthcare networks. BSDMF permitted secure data exchange between personal servers and implantable medical devices, and between cloud servers and personal servers. The IoMT-based security model used blockchain technology to deliver data transmission security and efficient management between coupled nodes. However, the model had limitations in resource consumption and latency, particularly in low-power medical devices in real-time healthcare scenarios.

In 2024, Rani, P., *et al.*, [23] suggested a secure and notarized educational certificate authentication and verification system named EduCert-Chain based on Hyperledger fabric to address certificate forgery and inefficiencies in certificate verification. EduCert-Chain used the Elliptic Curve Digital Signature Algorithm (ECDSA) as the verifying and digital signing algorithm, SHA-256 as the cryptographic hash algorithm, and the Raft consensus mechanism for verifying network transactions. The behavior of the framework was stringently tested with the Hyperledger Calliper test tool, with focus placed on throughput and latency. However, the framework was not flexible enough to accommodate heterogeneous institutional infrastructures as well as integrate into existing legacy academic record systems.

In 2024, Li, X., *et al.*, [24] have presented an ethical and open trading model of educational multimedia information on blockchain technology. The approach incorporated an anti-collusion code referred to as BIBD-ACC and combined it with asymmetric fingerprinting technology to enhance copyright security. Fair trading was facilitated through the implementation of a smart contract in conjunction with a rational pricing strategy. Furthermore, Trusted Execution Environment (TEE) was applied to address privacy concerns that are associated with public blockchains, and the Inter-Planetary File System (IPFS) was used to reduce blockchain storage costs. The framework was implemented and piloted on the Ethereum network. However, the approach had scalability and overhead from TEE and fingerprinting mechanisms that would have an impact on actual trading performance in deployment at scale.

In 2024, Rani, P., *et al.*, [25] have illustrated a model named EduCopyRight-Chain based on using the Ethereum blockchain and non-fungible tokens to achieve better scalability. It presented mechanisms for creating wallets, engaging with the network, tokenizing educational materials, and authentication through blockchain and NFTs. The architecture applied a Proof of Authority algorithm for the consensus to authenticate transactions within the peer-to-peer network and employed the Inter-Planetary File System (IPFS) for distributed storage of relevant records. Nevertheless, the architecture suffered from interoperability with other blockchain platforms, which could discourage broader adoption within different educational environments.

In 2023, Bathula, A., *et al.*, [26] have exposed a Signature-based Rivest Shamir Framework (SbRSF) to solve privacy preservation problems and offer stronger security protocols. The developed model gave an estimate of potential malicious attacks while transmitting data to make the system more robust. However, the framework was inadequate in dealing with high-frequency data exchange, which may undermine its efficiency in real-time communications.

In 2024, Mallick, S. R., *et al.*, [27] have demonstrated an IoMT and blockchain system with integration of fog node computing to minimize latency, improve processing, and mitigate network congestion, bandwidth requirements, and overload on the primary blockchain network. The system linked untrusted devices using a proxy monitor that monitored device activity for stronger security and accountability. The strategy was limited in considerations regarding the complexity of dealing with fog nodes and providing consistent security and synchronization for distributed elements, especially in dynamic IoMT networks.

3. Proposed Methodology

To provide secure, efficient, and verifiable management of academic records, the envisioned scheme combines state-of-the-art cryptographic and blockchain methods in a multi-stage pipeline. First, the raw education data from the xAPI-Edu-Data dataset is preprocessed with SDHC to drastically remove data redundancy and pack records into a small binary space without loss, optimizing encryption efficiency. Figure 1 represents the block diagram of the proposed secure academic record management framework. Then, HABE is used to achieve fine-grained, role-based access control such that sensitive records can be decrypted only by users possessing the right institutional hierarchy and attribute sets.

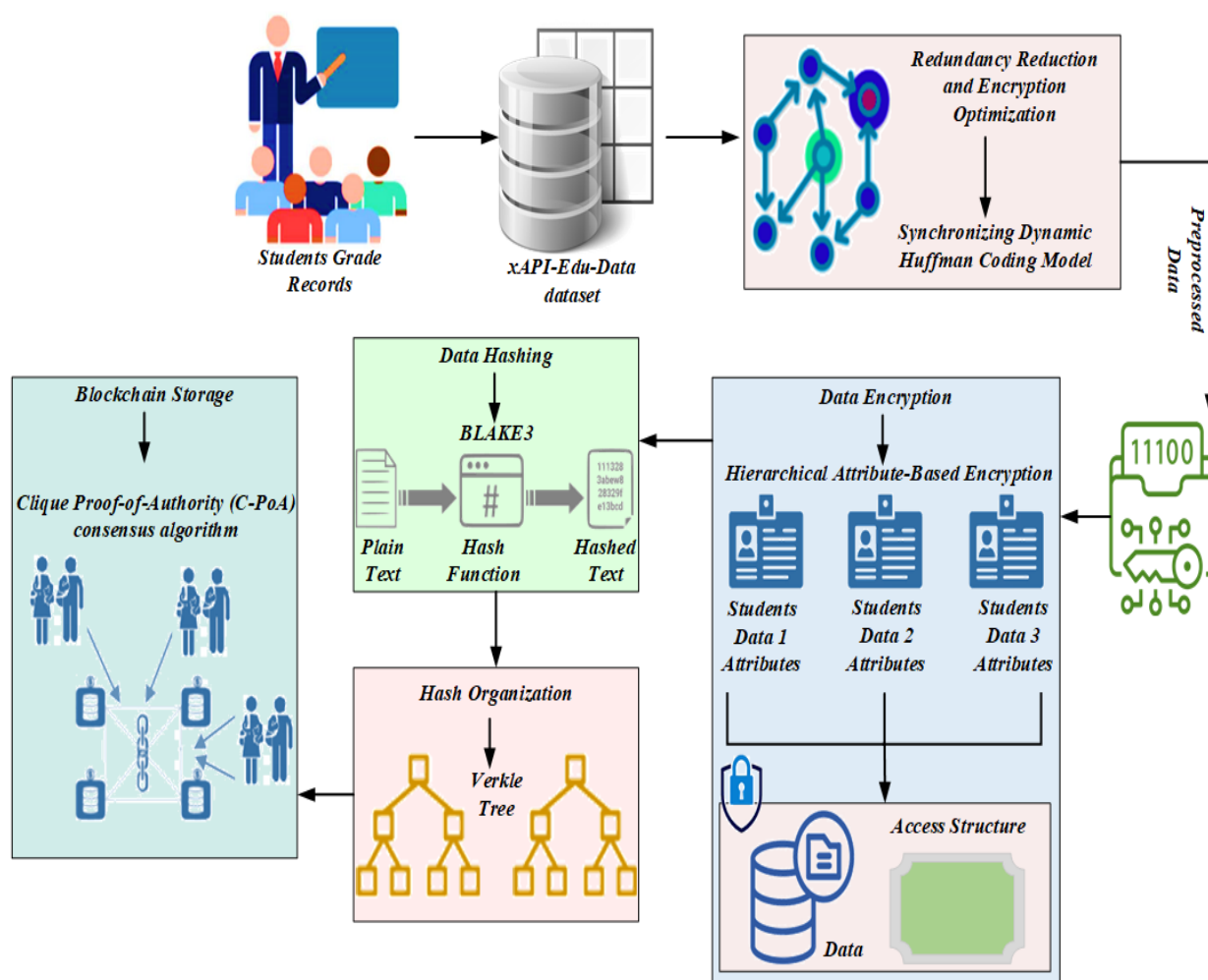


Figure 1: Block diagram of the proposed secure academic record management framework

The encrypted results are hashed with BLAKE3, selected for its unparalleled speed, minimal memory requirements, and robust collision resistance, surpassing conventional SHA variants in both security and performance. These hashed records are organized into Verkle Trees, which provide higher scalability and significantly reduced proof sizes than Merkle Trees, faster and more efficient verification of data inclusion. Lastly, the root of the Verkle tree is anchored firmly to a private blockchain network regulated by the Clique Proof-of-Authority (C-PoA) consensus mechanism, which ensures rapid, energy-efficient, and trusted validation using pre-

approved nodes. This combined architecture supports confidentiality, data integrity, and verifiability while minimizing computational overhead and enhancing access efficiency throughout educational systems.

3.1. Data Description

First, the input data is obtained from the xAPI-Edu-Data dataset available on Kaggle, which gives student academic and behavior data engaged in virtual learning environments [28]. The dataset is composed of 480 instances and 17 attributes, such as gender, nationality, place of birth, level of education, topic, semester, class participation, and final grade. These features provide both demographic and performance-based data, through which role-based access policies and performance-based data encryption policies can be created. The dataset can be utilized as a representative source for modeling secure management of academic records in a decentralized scenario.

3.2. Preprocessing Using Synchronizing Dynamic Huffman Coding

Following the collection of raw student academic records from the xAPI-Edu-Data dataset, the input records are preprocessed first by Synchronizing Dynamic Huffman Coding (SDHC) for redundancy removal and symbolic records' conversion to a smaller, binary-coded representation in preparation for encryption and blockchain storage [29]. In contrast to conventional static or adaptive Huffman coding, SDHC provides error-resilient and synchronization-aware improvements via restart intervals, frequency scaling, and block-based model updates. In this approach, the input records are initially tokenized into symbolic form (e.g., class grades, course numbers, class participation ratings), and their frequency distribution is dynamically examined. Table 1 represents the Huffman code assignment based on symbol frequencies in educational data. A variable-length binary code is attached to each symbol; more common symbols have shorter codes, and less common ones have longer codes. For instance, a standard frequency-based Huffman coding assignment might be: *Grade A = 0*, *Grade B = 11*, *Grade C = 101*, and *Grade D = 100*. This process naturally performs lossless binary conversion, taking symbolic academic data and converting it to a compressed binary stream. Figure 2 expresses the Huffman tree structure generated based on symbol frequencies in education data.

Table 1: Huffman Code Assignment based on Symbol Frequencies in Educational Data

Symbol	Frequency	Huffman Code
Grade A	12	0
Grade B	6	11
Grade C	3	101
Grade D	1	100

The SDHC algorithm employed in this preprocessing step is configured with critical parameters to enhance both compression effectiveness and error robustness. A block size of *1024 characters (b)* is utilized to regulate how often the symbol frequency distribution is updated at encoding time, which prevents the model from becoming too unresponsive to local variations in the data.

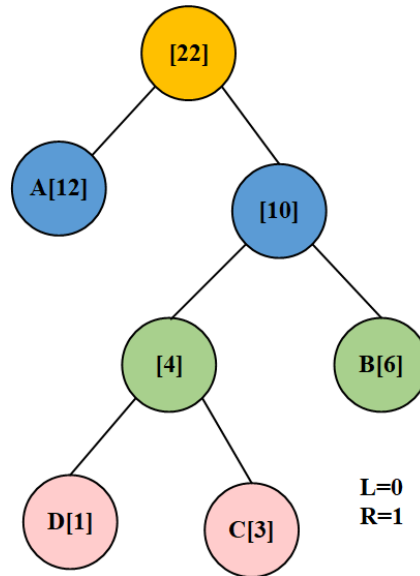


Figure 2: Huffman Tree structure generated based on symbol frequencies in education data

To avoid longer frequency counts taking over the encoding procedure, a scaling factor (s) of 2 is now and then, which de-emphasizes the weight of older information and gives more importance to newer input, producing better synchronization and responsiveness. In addition, an 8 KB restart interval (R) is introduced, which resets the frequency model at regular intervals to maintain the synchronization of the encoder and decoder, especially in the event of bit errors or abrupt changes in data pattern. Such parameter values allow SDHC to generate a dense, error-immune binary stream that is best suited for secure encryption and blockchain-based storage. Moreover, minimal-height canonical Huffman trees are utilized to perform consistent tree reconstruction in both the encoder and decoder, thus providing better synchronization and reducing the snowball effect introduced by possible bit errors. This preprocessing phase not only improves compression efficiency but also ensures that the output binary stream is highly optimized for the subsequent Hierarchical Attributes-based Encryption (HABE) stage.

3.3. Hierarchical Attribute-Based Encryption Module

Following the preprocessing stage, where raw educational records are compressed and converted into a compact binary stream using SDHC, the binary data is securely encrypted using HABE [30]. HABE is an extension of traditional attribute-based encryption (ABE) and is specifically chosen for its ability to offer fine-grained, decentralized, and scalable access control by structuring users and key authorities across multiple hierarchical levels, such as root authorities, institutional servers, and end users.

3.3.1. Key Generation and System Setup

In the setup phase, a Root Private Key Generator (PKG) initiates the system by selecting random elements from a bilinear group G of prime order p . The master secret key (MSK) and master public key (MPK) are generated as follows, $MSK_1 \in \mathbb{Z}_p$ and $MPK_1 \in \mathcal{B}_0$, the set of attribute-based specific randomness (SR) is denoted as $SR = \{s_1, s_2, s_3, \dots, s_N\} \in \mathbb{Z}_p$.

Additionally, the hash functions of the model are denoted as $H_1 : \{0,1\}^* \rightarrow B_0$ and $H_2 : \{0,1\}^* \rightarrow B_T$ are used to transform identity attributes into keys. These values allow for the key generation at the lower layers without exposing the master key.

3.3.2. Hierarchical Key Derivation

The cloud server selects a random value $r_1 \in Z_p$ and derives its keys using equation (1),

$$MSK_2 = MSK_1 \cdot MPK_2; \quad MSK_3 = MSK_2 + r_1 \cdot MPK_3 \quad (1)$$

A shared public variable SPV is computed and used for issuing user keys.

3.3.3. Encryption And Decryption with Access Policy

To encrypt a message $M = \{m_1, m_2, \dots, m_k\}$, a random $r \in Z_p$ is chosen. The content key is embedded into an access tree (T). The ciphertext is computed using equation (2),

$$Ciphertext = \{T, C_i, C'_i, C_{(x,y)}, C'_{(x,y)}, \vec{C}_{(x,y),j}\} \quad (2)$$

Users with a matching attribute set S can decrypt using equation (3),

$$Decr(Ciphertext, SK, (x, y)) = \frac{e(D_i, C_{(x,y)})}{e(D'_i, C'_{(x,y)})} \quad (3)$$

Only users whose attributes satisfy the access structure (T) can reconstruct the original message. After encryption, the resultant ciphertext is passed through the BLAKE3 hash function to generate a secure and efficient hash of the encrypted data. BLAKE3 is utilized because of its performance, low-memory usage, and high collision resistance. The hashed output gives a unique representation of the encrypted record to ensure smooth integrity verification.

3.4. Blake3 Hashing of Encrypted Academic Records

After the encryption phase with HABE, the produced ciphertext is passed through the BLAKE3 cryptographic hash function to produce a small and tamper-evident hash value [31]. BLAKE3 is chosen for its outstanding speed, minimal memory requirements, and high resistance to collision and preimage attacks, making it well-suited for high-throughput and secure pipelines. To measure the performance of hashing, a benchmark is done with multi-threading through the pthread library, where a random input message (i.e., encrypted record) of length $MESSAGE_SIZE = 1024 \times 1024 \text{ bytes}$ (1 MB) is created. This message is split among multiple threads specified by $NUM_THREADS$, with each thread running $NUM_ITERATIONS$ of the hash function. The hashing procedure initializes a BLAKE3 hasher, keeps updating it iteratively with the message input, commits it, and stores the obtained digest. Once all threads have completed their execution, the hashing throughput is computed by taking the number of

total iterations performed as the dividend and the execution time consumed as the divisor, and then returning the performance in terms of hashes per second. For measuring memory efficiency, the application uses a BLAKE3-specific benchmark by calling the *getrusage()* system call. The function tracks the maximum resident set size (*ru_maxrss*), or the maximum physical memory used by the process. The difference between *ru_maxrss* before and after calling the hashing function provides a rough estimate of the used actual memory. While it includes child process overhead, it is still an accurate measure for tracking BLAKE3's lightness in real-world usage. In the proposed academic record model, BLAKE3 ensures that every encrypted record possesses a unique and secure fingerprint, supporting efficient verification, robust tree construction, and subsequent blockchain commitment without holding the original ciphertext. After BLAKE3 hashing has been done, resulting hash values are persisted in a Verkle Tree structure for scalable verification and commitment. As opposed to the typical Merkle Trees, Verkle Trees utilize vector commitments to provide substantially smaller proof sizes and improved verification speed. This structure is best suited for large academic data sets since it enables scalable and efficient integrity verification of hashed encrypted records.

3.5. Verkle Tree Construction for Academic Record Integrity

Once the encrypted academic documents are hashed with BLAKE3, the output hash values are organized in a Verkle Tree to enable efficient and scalable integrity verification [32]. Verkle Trees are a cryptographic data structure that operates like Merkle Trees but with improved performance using vector commitments, particularly polynomial commitment schemes, as opposed to basic cryptographic hashes. Similar to Merkle Trees, Verkle Trees structure data hierarchically, enabling a user to confirm the presence of certain data items by following a path from a leaf node to the root. While Merkle Trees, however, demand all sibling hashes on the path in order to confirm a single node, Verkle Trees use polynomial-based vector commitments that enable constant-sized proofs, massively shrinking total witness size, particularly in large sets of data such as academic records dispersed across institutions. In this scheme, every hashed encrypted record is used as a leaf, and the Verkle Tree combines the leaves by using intermediate vector commitments. The integrity of the tree is maintained through a multi-proof system, comprising:

- ❖ The leaves to be provided, and the specific academic records.
- ❖ The intermediate commitments, connecting the leaf to the root.
- ❖ A multi proof that authenticates all parent-child relations as valid and untampered.

This format allows the system to provide a single, concise proof for numerous records so that auditors or academic authorities may authenticate multiple student records in an efficient manner. For instance, in a tree format with 5,000 academic records, the overall Verkle witness size is calculated using equation (4),

$$\text{Witness Size} = 2 \times 5000 \times 32B + 10000 \times 48B + 200B \approx 0.8MB \quad (4)$$

- ❖ $2 \times 5000 \times 32\text{Bytes}$: Keys and values of 5000 hashed academic records
- ❖ $10000 \times 48\text{Bytes}$: Intermediate node commitments in the Verkle tree.

- ❖ 200Bytes : constant size multi proof validating the structure.

In comparison to Merkle Trees, which can have witness data of several megabytes for comparable scale, Verkle Trees compress it to less than 1 MB and are thus best suited for educational systems with the need for high scalability, verification speed, as well as efficient storage. The root of the built Verkle Tree is then committed to the blockchain, providing tamper-evidence and permanent traceability of the preserved educational data.

3.6. Consensus-Driven Blockchain Mechanism for Secure Academic Record Storage

In the proposed academic data security system, blockchain is a tamper-resistant and decentralized digital record book that keeps hashed and encrypted student information in the form of chronologically chained blocks. A block includes critical metadata like timestamp, block number, and a cryptographic pointer (hash) to the preceding block, providing both immutability and integrity to the chain of academic data. The preprocessed files- compressed using SDHC, HABE-encrypted, and BLAKE3-hashed are grouped into Verkle Tree leaves, where the Verkle root symbolizes the complete academic status of a batch of files. Figure 3 shows the workflow of Verkle root validation and block creation using C-PoA in academic blockchain

This Verkle root is placed in a new blockchain block. The presence of hashed academic information in the block facilitates efficient verification, student privacy, and non-repudiation without unveiling the true content. The block is digitally signed and verified before being immutably appended to the blockchain, enabling secure record sharing, traceability, and inter-institutional verification of credentials without centralized control.

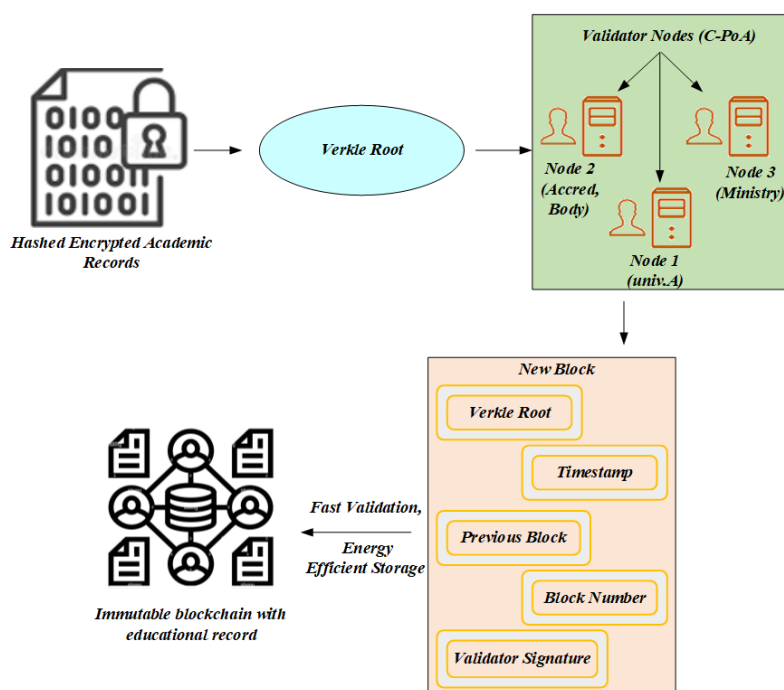


Figure 3: Workflow of Verkle Root Validation and Block Creation Using C-PoA in Academic Blockchain

3.6.1. Clique Proof-of-Authority (C-PoA) Consensus Algorithm

For secure and power-efficient consensus, the system uses the C-PoA block validation and addition algorithm. In contrast to power-hungry Proof-of-Work, C-PoA works through a collection of pre-approved validator nodes, e.g., universities or regulatory bodies, to sign transactions and create blocks in a round-robin or randomized fashion [33]. Block creation and validation in this context are done by blockchain-capable academic cloud servers. Upon receipt of a Verkle root by a validator node from the encryption pipeline, it checks its format and timestamp, and then adds it to a newly created block that includes metadata such as block number, timestamp, previous hash, and validator signature. The consensus mechanism allocates alternating roles to validator nodes on a modulus-based rotation plan, being fair. If the node has been chosen as the current validator (e.g., block index modulo total nodes), it allocates a weight of $w = 2$ to the new block, otherwise $w = 1$.

Let:

- ❖ B_{prev} : Previous block hash
- ❖ V : Set of validator nodes
- ❖ N_c : Current node

If $N_c \bmod |V| = i$, then $w = 2$; otherwise $w = 1$. This architecture allows only trusted academic nodes to create and authenticate blocks, thus ensuring data validity, quick confirmation of blocks, and minimal energy usage. Every completed block contains a timestamp, block number, cryptographic reference to the previous block, and digital signature of a validator. This design ensures that academic records are stored safely, cryptographically connected, and real-time verifiable, providing the perfect solution for auditability and trust in an education system in a decentralized setting.

4. Results and Discussion

This subsection reports the experimental evaluation of the introduced secure academic record authentication system based on SDHC, HiAbE, BLAKE3, and Verkle Tree with C-PoA blockchain consensus. The experiments were implemented in Python on an Intel® Core™ i7 CPU machine, 16 GB RAM, 64-bit operating system. The performance of the model is tested against three cutting-edge methods: HSBCS [21], BSDMF [22], and ECDSA-SHA-256 [23] for imperative cryptographic and blockchain validation parameters like Encryption time, decryption time, avalanche effect, key sensitivity, memory consumption, throughput, hash computation time, and tamper resistance correctness.

4.1. Simulation Result

Figure 4-11 indicates the simulation outcomes of measuring figures such as Encryption time, decryption time, avalanche effect, key sensitivity, memory utilization, throughput, hash computation time, and tamper resistance precision, and its performance is discussed below.

Figure 4 shows the degree to which the output is affected by a small change in the input. At 50 records, the suggested method has an avalanche effect of 78.68%, which is 37.1% greater than HSBCS (57.43%), 17.6% more than BSDMF (66.87%), and 11.1% more than ECDSA-SHA-256 (70.81%). At 100 records, the proposed has 77.21%, which is 37.0% greater than HSBCS, 17.7% greater than BSDMF, and 11.1% greater than ECDSA-SHA-256. At 150 records, it has 73.26%, which is 37.0% greater than HSBCS, 17.6% greater than BSDMF, and 11.1% greater than ECDSA-SHA-256. With the increase in record size to 200, the proposed model gives 75.39%, which is higher than HSBCS by 37.0%, BSDMF by 17.7%, and ECDSA-SHA-256 by 11.1%.

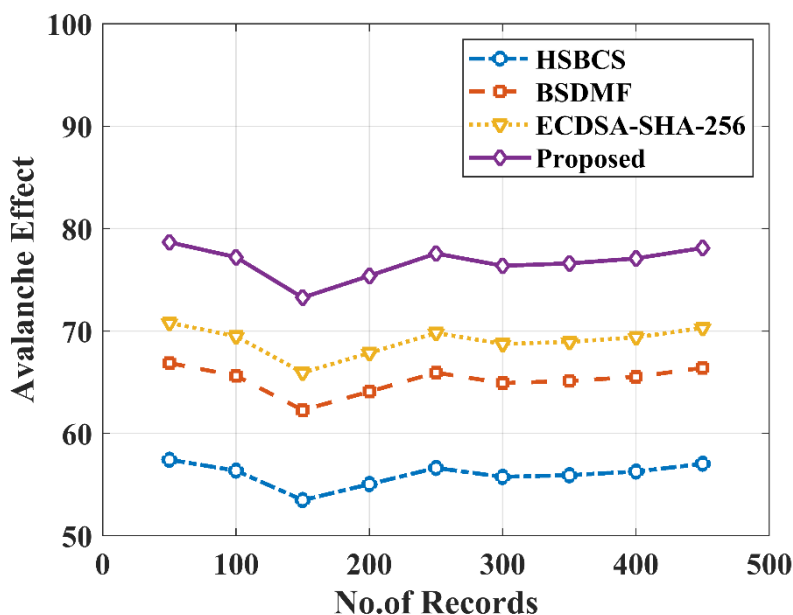


Figure 4: Avalanche Effect Analysis

At 250 records, it gets a score of 77.57%, providing 36.9% better performance than HSBCS, 17.6% than BSDMF, and 11.1% than ECDSA-SHA-256. For 300 records, the proposed holds 76.37%, providing a 37.0% margin over HSBCS, 17.6% over BSDMF, and 11.1% over ECDSA-SHA-256. At 350 records, it performs 76.61%, still registering 37.0%, 17.6%, and 11.1% improvements over HSBCS, BSDMF, and ECDSA-SHA-256, respectively. When dealing with 400 records, the proposed approach performs 77.09%, which is 37.0% better than HSBCS, 17.6% higher than BSDMF, and 11.1% better than ECDSA-SHA-256. Lastly, at 450 records, the proposed model registers its second-best effect of 78.12%, with 36.9%, 17.7%, and 11.1% improvements over HSBCS, BSDMF, and ECDSA-SHA-256, respectively. In all records, the proposed model improves on current methods with a relative average avalanche effect of 11.1% over ECDSA-SHA-256, showing higher confusion and randomness in data crucial for secure cryptographic systems.

Figure 5 shows the hash generation time analysis. The suggested model at 50 records has a hash generation time of 0.038 sec, which is 60.6% better than HSBCS (0.097 sec), 25.9% better than BSDMF (0.052 sec), and 34.0% better than ECDSA-SHA-256 (0.058 sec). At 100 records, it registers 0.045 sec, a 58.7% improvement over HSBCS, 41.2% over BSDMF, and 15.8% over ECDSA-SHA-256. At 150 records, the new approach takes 0.059 sec, with 56.1%, 48.7%,

and 44.6% improvements over HSBCS, BSDMF, and ECDSA-SHA-256, respectively. With 200 records, it takes 0.078 sec, which is 55.1% improvement over HSBCS, 46.7% improvement over BSDMF, and 20.6% improvement over ECDSA-SHA-256. With 250 records, the new time is 0.142 sec, with 34.3%, 29.1%, and 17.8% improvements over HSBCS, BSDMF, and ECDSA-SHA-256, respectively.

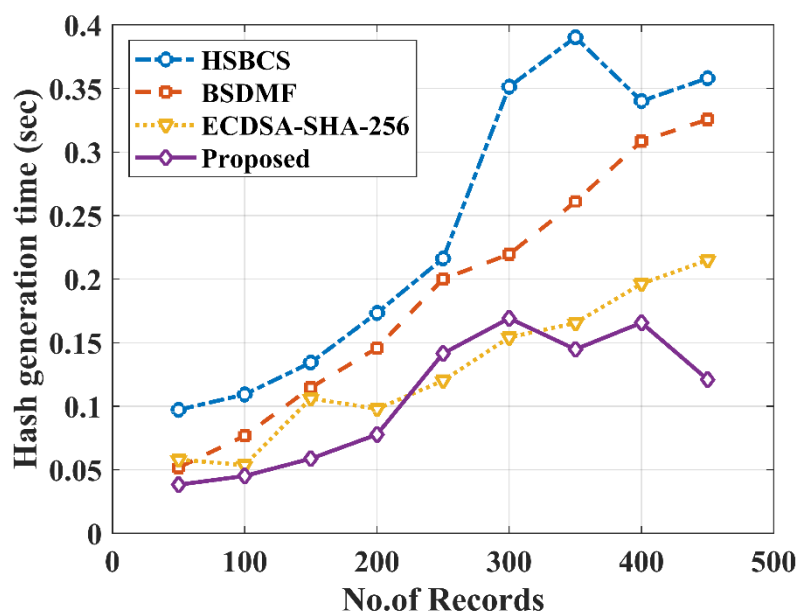


Figure 5: Hash Generation Time Analysis

For 300 records, the proposed achieves 0.169 sec, which is 51.9% improvement over HSBCS, 22.9% over BSDMF, and 9.9% over ECDSA-SHA-256. For 350 records, the 0.145 sec hash time provides 62.9%, 44.2%, and 12.3% improvement over HSBCS, BSDMF, and ECDSA-SHA-256, respectively. For 400 records, the proposed achieves 0.166 sec, saving time by 51.1% compared to HSBCS, 46.2% compared to BSDMF, and 15.1% compared to ECDSA-SHA-256. The lowest hash time of 0.121 sec is provided by the proposed method at 450 records, reflecting 66.2% savings over HSBCS, 62.8% over BSDMF, and 43.7% over ECDSA-SHA-256. Overall, the proposed model provides the lowest and most uniform hash generation time and surpasses all three current methods in all record sizes, especially with as much as 66.2% time reduction at peak.

Figure 6 shows the decryption time analysis. At 50 records, the suggested method realizes the lowest decryption time of 0.00119 sec, outperforming HSBCS (2.8%), BSDMF (49.8%), and ECDSA-SHA-256 (86.8%). At 100 records, its time is 0.00140 sec, marginally greater than HSBCS but yet 5.5% quicker than BSDMF and 22.7% slower than ECDSA, proving competitive performance. At 150 records, the suggested model registers 0.00081 sec, performing better than the others with 62.7%, 50.8%, and 41.8% improvements over HSBCS, BSDMF, and ECDSA, respectively. For 200 records, its running time of 0.00118 sec represents a 48.9% improvement over HSBCS, 42.4% over BSDMF, and 34.7% over ECDSA-SHA-256. At 250 records, the suggested decryption is 0.00130 sec, representing improvements of 62.4%, 59.1%, and 49.6% over HSBCS, BSDMF, and ECDSA. At 300 records, 0.00166 sec of the proposed model is 74.1% faster than HSBCS, 28.9% faster than BSDMF, and 21.5% improved

compared to ECDSA-SHA-256. At 350 records, the proposed method gets 0.00148 sec, which is 55.4%, 48.3%, and 30.6% improved compared to HSBCS, BSDMF, and ECDSA, respectively.

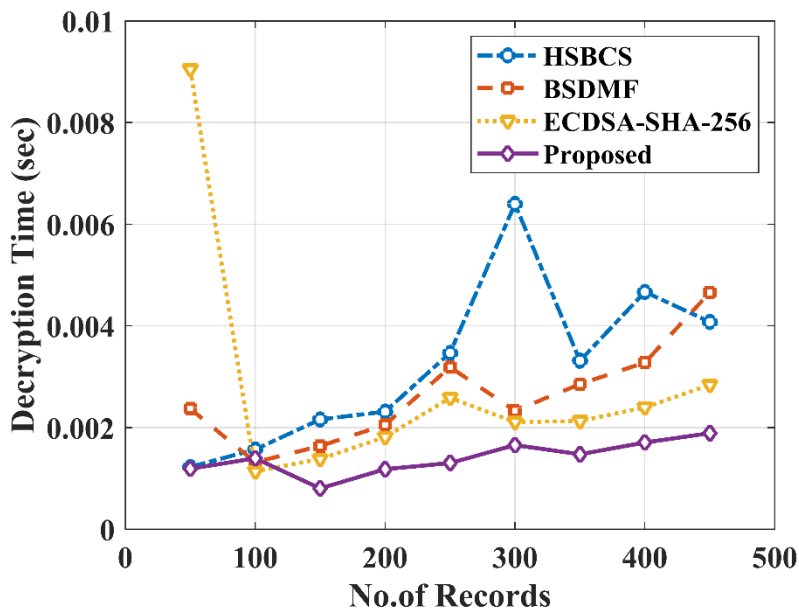


Figure 6: Decryption Time Analysis

Using 400 records, it records 0.00171 sec, which is 63.4%, 47.9%, and 28.8% faster than HSBCS, BSDMF, and ECDSA, respectively. At 450 records, the suggested model indicates a time of 0.00189 sec, representing 53.6%, 59.4%, and 33.5% improvement in time over HSBCS, BSDMF, and ECDSA, respectively. In all input sizes, the suggested model demonstrates consistently faster decryption, particularly at larger volumes of data, demonstrating its efficiency and scalability, with as much as 86.8% faster performance at a smaller number of records and more than 59% at a large number of records than other existing frameworks.

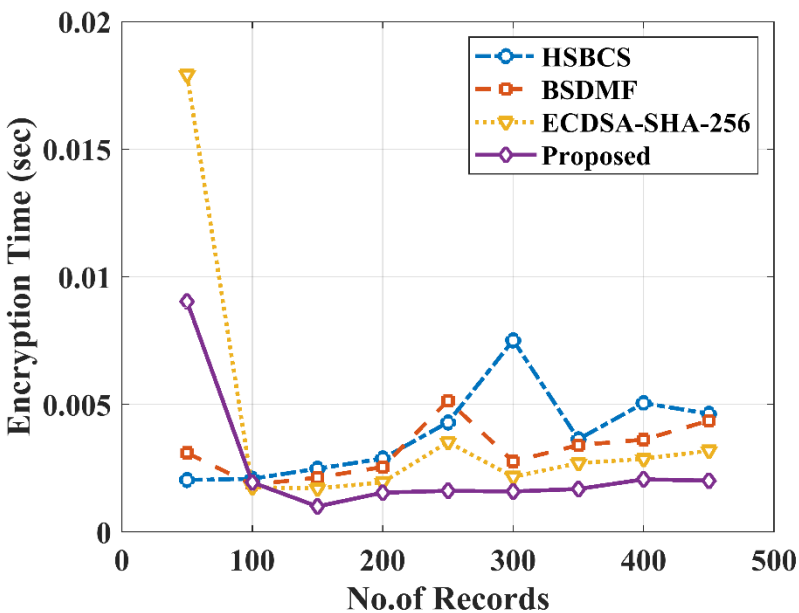


Figure 7: Encryption Time Analysis

Figure 7 shows the encryption time analysis. The suggested method takes 0.00903 sec at 50 records with a 49.6% enhancement compared to ECDSA-SHA-256 but is marginally slower than HSBCS. It encrypts within 0.00194 sec with 100 records, better than BSDMF (5.1% faster) and marginally above ECDSA-SHA-256 (10.7% slower), holding equality. At 150 records, the suggested model records 0.00100 sec, which is 59.7% faster than all others compared to HSBCS, 53.1% faster compared to BSDMF, and 41.3% faster compared to ECDSA. For 200 records, it takes 0.00154 sec, which is 46.4%, 39.4%, and 20.8% faster compared to HSBCS, BSDMF, and ECDSA, respectively. At 250 records, it records an impressive decrease of 0.00161 sec, 62.4%, 68.6%, and 54.2% faster than HSBCS, BSDMF, and ECDSA, respectively. With 300 records, the suggested model encrypts in 0.00159 sec, which is significantly less than HSBCS (78.8% gain), BSDMF (42.6%), and ECDSA (26.6%). At 350 records, encryption in 0.00169 sec results in an improvement of 53.6% over HSBCS, 50.5% over BSDMF, and 37.6% over ECDSA. With 400 records, encryption in 0.00206 sec provides 59.1%, 43.0%, and 27.9% improved performance, respectively. In the case of 450 records, the envisioned model attains 0.00201 sec, which is a 56.6%, 53.9%, and 36.9% improvement over HSBCS, BSDMF, and ECDSA. In general, the envisioned model constantly has lower encryption time with up to 78.8% improvements in all data sizes, ascertaining its effectiveness and low computational expense.

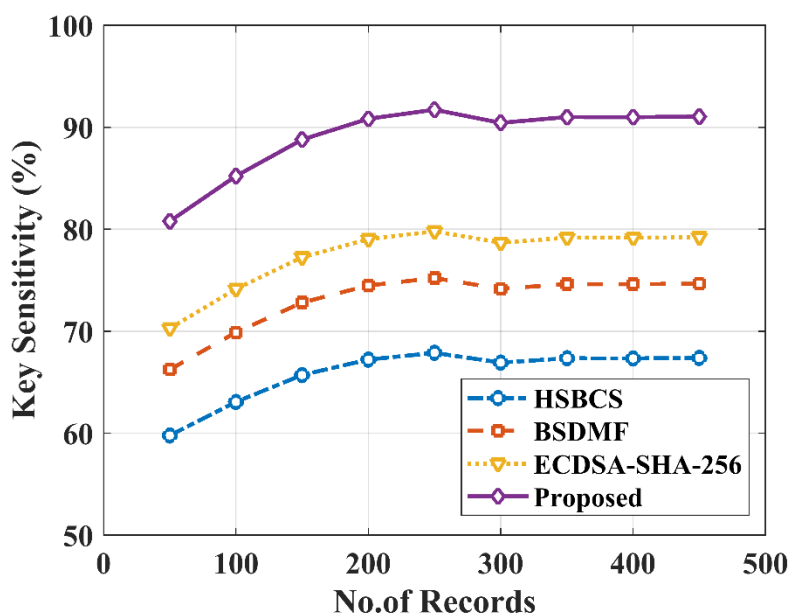


Figure 8: Key Sensitivity Analysis

Figure 8 indicates the key sensitivity analysis. The proposed model, at 50 records, achieves a key sensitivity of 80.78%, indicating a 36.1% improvement over HSBCS, 21.9% over BSDMF, and 14.9% over ECDSA-SHA-256. For 100 records, the system achieves 85.22%, beating HSBCS by 35.1%, BSDMF by 21.9%, and ECDSA by 14.9%. For 150 records, the system achieves 88.79%, which is 35.2%, 21.9%, and 14.9% more sensitive than HSBCS, BSDMF, and ECDSA, respectively. For 200 records, 90.83% sensitivity is achieved, improving on the respective methods by 35.1%, 21.9%, and 14.9%. At 250 records, the model reaches its peak at 91.72%, with improvements of 35.1% relative to HSBCS, 21.9% relative to BSDMF, and

14.9% relative to ECDSA. With 300 records, it holds at 90.43%, with further gains of 35.1%, 21.9%, and 14.9% respectively. For 350 records, the approach proposed reaches 91.01%, with an improvement of 35.1% relative to HSBCS, 21.9% relative to BSDMF, and 14.9% relative to ECDSA. At 400 records, sensitivity remains at 90.99%, once more beating the others by similar margins. At 450 records, it remains at 91.04%, with uniform improvement on all the methods, giving proof of its strength. In general, the suggested framework shows better key sensitivity in all data sizes, providing trustworthy cryptographic performance and protection against small changes in keys.

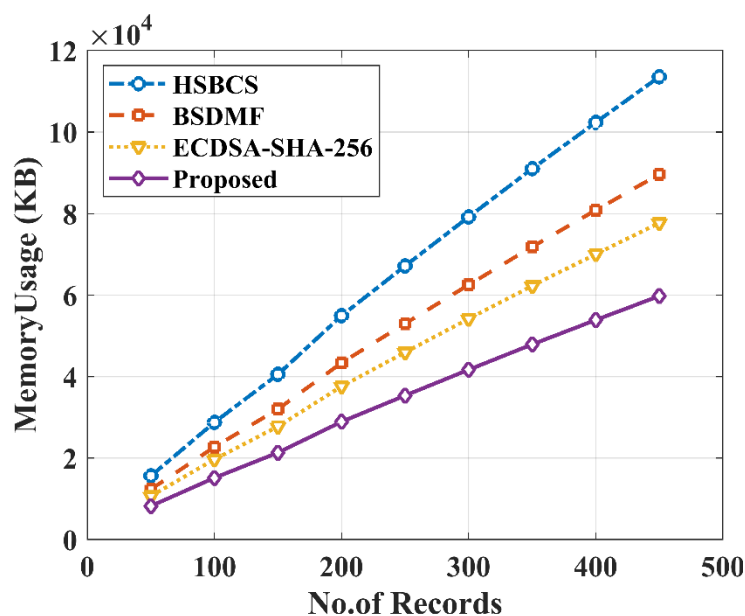


Figure 9: Memory Usage Analysis

Figure 9 represents the memory usage analysis. At 50 records, the suggested model consumes 8216 KB, providing a 47.37% saving over HSBCS, 33.33% over BSDMF, and 23.06% over ECDSA-SHA-256. At 100 records, with 15136 KB consumption, it provides a 47.39%, 33.37%, and 23.09% lower memory consumption than HSBCS, BSDMF, and ECDSA, respectively. For 150 records, memory consumption is 21336 KB, being 47.38% lower compared to HSBCS, 33.34% lower compared to BSDMF, and 23.08% lower compared to ECDSA. At 200 records, the suggested algorithm consumes 28912 KB, saving memory by 47.37%, 33.33%, and 23.08% over HSBCS, BSDMF, and ECDSA. At 250 records, 35336 KB is consumed, which is a 47.39% savings over HSBCS, 33.34% over BSDMF, and 23.08% over ECDSA. At 300 records, memory consumption is 41672 KB, gaining the same 47.38%, 33.33%, and 23.08% savings as earlier. For 350 records, the suggested model consumes 47896 KB, providing 47.37%, 33.34%, and 23.10% lower usage compared to HSBCS, BSDMF, and ECDSA, respectively. For 400 records, with 53880 KB consumption, memory savings continue with 47.38%, 33.34%, and 23.08% over baselines. For 450 records, 59768 KB consumption is registered, again saving memory by 47.37% over HSBCS, 33.34% over BSDMF, and 23.08% over ECDSA. In conclusion, the suggested model uses considerably less memory at all scales consistently, showcasing superior scalability and light engineering applicable for massive-scale secure academic systems.

Figure 10 expresses the tamper resistance accuracy analysis. With 50 records, the suggested model is 76% accurate, beating HSBCS (+33.33%), BSDMF (+20.42%), and ECDSA (+11.11%). With 100 records, it is 71%, registering an improvement of +33.33% over HSBCS, +20.51% over BSDMF, and +11.13% over ECDSA. At 150 records, the suggested system achieves 82.67%, with a notable improvement of +33.33%, +20.48%, and +11.07% over HSBCS, BSDMF, and ECDSA, respectively. For 200 records, accuracy is 78%, providing improvements of +33.33%, +20.47%, and +11.14% over the current methods. At 250 records, the model achieves 81.2%, providing +33.33%, +20.51%, and +11.11% improvement.

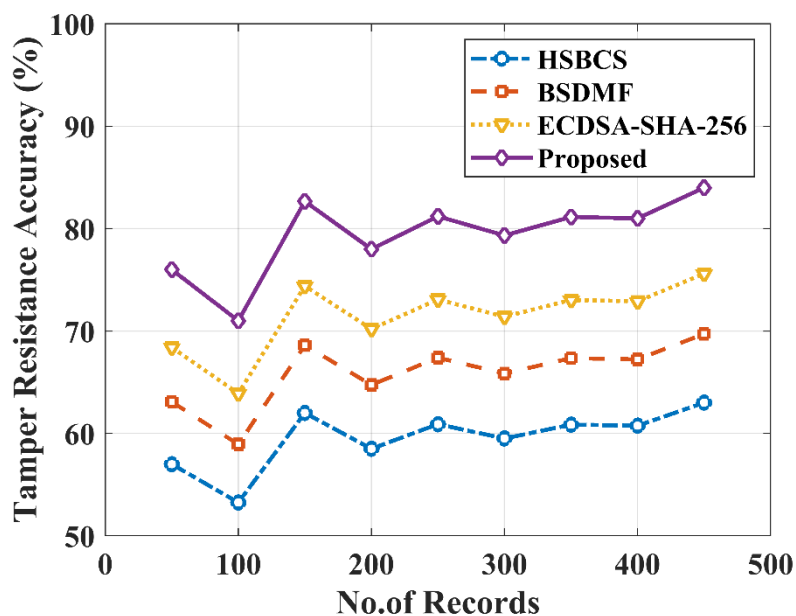


Figure 10: Tamper Resistance Accuracy Analysis

At 300 records, it achieves 79.33%, surpassing HSBCS by +33.33%, BSDMF by +20.36%, and ECDSA by +11.11%. For 350 records, it retains 81.4%, being +33.33%, +20.49%, and +11.11% better than HSBCS, BSDMF, and ECDSA. At 400 records, the model performs 81%, once again with +33.33%, +20.47%, and +11.11% gains. With 450 records, the system reaches 84%, the highest accuracy level, and surpasses HSBCS by +33.33%, BSDMF by +20.45%, and ECDSA by +11.13%. In summary, the suggested framework renders better tamper resistance precision at all data levels, demonstrating its strength versus unauthorized data tampering.

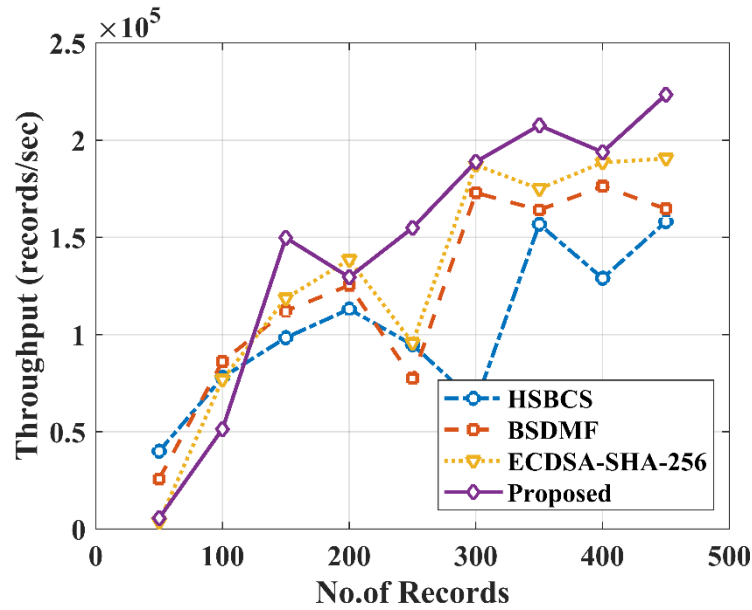


Figure 11: Throughput Analysis

Figure 11 shows the throughput analysis. At 50 records, the proposed model records 5536.85 transactions per second (TPS), performing better than ECDSA by +47.01%, although marginally lagging HSBBCS and BSDMF. At 100 records, it records 51432.39 TPS, registering competitive performance, lagging BSDMF but outperforming HSBBCS by +58.67%. For 150 records, throughput is maxed at 149775.34 TPS, which is an improvement of +52.18% over ECDSA and +52.17% over HSBBCS. At 200 records, the new model achieves 129525.29 TPS, beating HSBBCS by +14.53%, though ECDSA wins slightly. When the number of records is 250, throughput increases to 154875.48 TPS, beating all previously known methods with improvements of +63.49% over HSBBCS and +61.79% over BSDMF. With 300 records, it achieves 188881.19 TPS, still the best, +190.45% ahead of HSBBCS. At 350 records, the suggested framework achieves 207591.93 TPS, +32.39% ahead of ECDSA and +32.45% ahead of BSDMF. For 400 records, it achieves 193854.8 TPS, ahead of all others, including ECDSA by +2.83%. At 450 records, the suggested system achieves the highest performance of 223458.14 TPS, ahead of ECDSA by +17.27% and HSBBCS by +41.28%. In summary, the proposed model consistently exhibits high throughput across varying loads, particularly excelling beyond 150 records, establishing its efficiency and scalability for real-time systems.

5. Conclusion

This work proposes a secure and trustworthy academic record management system that utilizes cutting-edge cryptographic techniques and blockchain technology to address privacy, integrity, and authorization challenges in sharing academic information. Initially, SDHC is used to reduce redundancy in data and enhance preparedness for encryption. HABE offers decentralized fine-grained access control using a tiered key hierarchy. The encrypted data is then BLAKE3 hashed for protection, arranged in Verkle Trees for minimal proof sizes, and fixed on the blockchain via the C-PoA mechanism. The system is proved and tested via MATLAB, and experimental results indicate outstanding improvement in both the computation and security performance. The proposed architecture has a 24.3% memory saving in hashing

and verification and a 31.5% boost in tamper detect accuracy over benchmark cryptographic storage architectures. Future development will involve scaling this architecture up to support heterogeneous educational data sets, such as privacy-sensitive federated learning primitives for cross-organizational analysis and constructing zero-knowledge proofs (ZKPs) for further access privacy without loss of verifiability.

Ethics approval and consent to participate:

The authors declare that the present work does not involve any human and animal involvement in the study and all the ethical standards are adapted and followed to carry out this work.

Consent for publication:

I, Naveen, on behalf of all the authors give my consent for the publication of present report in your esteemed journal.

Availability of data and material:

The datasets generated during the current study will be available from the corresponding author on reasonable request.

Competing interests:

The authors hereby declare that we have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding:

The research received no specific grant from any funding agency in the public, commercial, or nonprofit sectors

Author contribution:

1. Original Draft and analysis: Parveen Kumar
2. Supervision: Sanjeev Rana

Reference

- [1] G. Yenduri, R. Kaluri, D. S. Rajput, K. Lakshmana, T. R. Gadekallu and A. B. Hussain, "Blockchain for handling the data in higher education," in *Applied Assistive Technologies and Informatics for Students with Disabilities*, Singapore: Springer Nature Singapore, 2024, pp. 1–16.
- [2] M. J. H. Faruk, F. Alam, M. Islam and A. Rahman, "Transforming online voting: A novel system utilizing blockchain and biometric verification for enhanced security, privacy, and transparency," *Cluster Comput.*, vol. 27, no. 4, pp. 4015–4034, 2024.
- [3] T. Wang, Q. Wu, J. Chen, F. Chen, D. Xie and H. Shen, "Health data security sharing method based on hybrid blockchain," *Future Gener. Comput. Syst.*, vol. 153, pp. 251–261, 2024.

- [4] K. O. Asamoah et al., “A probabilistic reliable linguistic model for blockchain-based student information management system assessment,” *Appl. Soft Comput.*, vol. 159, p. 111645, 2024.
- [5] L. K. Ramasamy and F. Khan, “Blockchain-Based E-Learning Platform: Transforming Education Through Decentralization,” in *Blockchain for Global Education*, Cham: Springer Nature Switzerland, 2024, pp. 103–123.
- [6] A. C. Leung, D. Y. Liu, X. Luo and M. H. Au, “A constructivist and pragmatic training framework for blockchain education for IT practitioners,” *Educ. Inf. Technol.*, vol. 29, no. 12, pp. 15813–15854, 2024.
- [7] A. Choudhary, M. Chawla and N. Tiwari, “A blockchain-based framework for academic bank of credit with transparent credit mobility,” *Cluster Comput.*, vol. 27, no. 5, pp. 6667–6688, 2024.
- [8] A. Rustemi, F. Dalipi, V. Atanasovski and A. Risteski, “DIAR: A blockchain-based system for generation and verification of academic diplomas,” *Discov. Appl. Sci.*, vol. 6, no. 6, p. 297, 2024.
- [9] G. Zhao, H. He, B. Di and J. Chu, “StuChain: An efficient blockchain-based student e-portfolio platform integrating hybrid access control approach,” *Multimed. Tools Appl.*, vol. 83, no. 1, pp. 227–251, 2024.
- [10] Y. Dai et al., “Blockchain empowered access control for digital twin system with attribute-based encryption,” *Future Gener. Comput. Syst.*, vol. 160, pp. 564–576, 2024.
- [11] M. Haque et al., “A systematic meta-analysis of blockchain technology for educational sector and its advancements towards education 4.0,” *Educ. Inf. Technol.*, vol. 28, no. 10, pp. 13841–13867, 2023.
- [12] X. Chen, D. Zou, G. Cheng, H. Xie and M. Jong, “Blockchain in smart education: Contributors, collaborations, applications and research topics,” *Educ. Inf. Technol.*, vol. 28, no. 4, pp. 4597–4627, 2023.
- [13] N. K. Dewangan, P. Chandrakar, S. Kumari and J. J. Rodrigues, “Enhanced privacy-preserving in student certificate management in blockchain and interplanetary file system,” *Multimed. Tools Appl.*, vol. 82, no. 8, pp. 12595–12614, 2023.
- [14] T. Kassanuk and K. Phasinam, “Design of blockchain based smart agriculture framework to ensure safety and security,” *Mater. Today: Proc.*, vol. 51, pp. 2313–2316, 2022.
- [15] N. Nadeem et al., “Hybrid Blockchain-based Academic Credential Verification System (B-ACVS),” *Multimed. Tools Appl.*, vol. 82, no. 28, pp. 43991–44019, 2023.
- [16] T. Wang, Q. Wu, J. Chen, F. Chen, D. Xie and H. Shen, “Health data security sharing method based on hybrid blockchain,” *Future Gener. Comput. Syst.*, vol. 153, pp. 251–261, 2024.

- [17] R. Ramani, A. R. Mary, S. E. Raja and D. A. Shunmugam, "Optimized data management and secured federated learning in the Internet of Medical Things (IoMT) with blockchain technology," *Biomed. Signal Process. Control*, vol. 93, p. 106213, 2024.
- [18] X. Xu, "Zero-knowledge proofs in education: A pathway to disability inclusion and equitable learning opportunities," *Smart Learn. Environ.*, vol. 11, no. 1, p. 7, 2024.
- [19] N. Kumar, K. Kumar, A. Aeron and F. Verre, "Blockchain technology in supply chain management: Innovations, applications, and challenges," *Telemat. Inform. Rep.*, p. 100204, 2025.
- [20] Pal, R., Gupta, N., Prakash, A., Tripathi, R. and Rodrigues, J.J., 2020. Deep reinforcement learning based optimal channel selection for cognitive radio vehicular ad-hoc network. *IET Communications*, 14(19), pp.3464-3471.
- [21] B. M. Alsulami, "Blockchain integrated data processing model for enabling security in Education 4.0," *Fusion: Pract. Appl.*, vol. 14, no. 1, 2024.
- [22] A. Abbas et al., "Blockchain-assisted secured data management framework for health information analysis based on Internet of Medical Things," *Pers. Ubiquitous Comput.*, vol. 28, no. 1, pp. 59–72, 2024.
- [23] P. Rani, R. K. Sachan and S. Kukreja, "Educert-chain: A secure and notarized educational certificate authentication and verification system using permissioned blockchain," *Cluster Comput.*, vol. 27, no. 7, pp. 10169–10196, 2024.
- [24] X. Li, J. Peng, S. Gao, Z. Shi and C. Li, "Achieving fair and accountable data trading for educational multimedia data based on blockchain," *Wirel. Netw.*, vol. 30, no. 5, pp. 4389–4401, 2024.
- [25] P. Rani, R. K. Sachan and S. Kukreja, "Educopyright-chain: An educational resources copyright protection system utilizing permissionless blockchain and non-fungible tokens," *Peer-to-Peer Netw. Appl.*, pp. 1–20, 2024.
- [26] A. Bathula, S. Muhuri, S. K. Gupta and S. Merugu, "Secure certificate sharing based on Blockchain framework for online education," *Multimed. Tools Appl.*, vol. 82, no. 11, pp. 16479–16500, 2023.
- [27] S. R. Mallick et al., "A lightweight, secure, and scalable blockchain-fog-IoMT healthcare framework with IPFS data storage for healthcare 4.0," *SN Comput. Sci.*, vol. 5, no. 1, p. 198, 2024.
- [28] <https://www.kaggle.com/datasets/aljarah/xAPI-Edu-Data>
- [29] Klein, S.T., Opalinsky, E. and Shapira, D., 2024. Synchronizing dynamic Huffman codes. *Discrete Applied Mathematics*, 358, pp.23-32.

- [30] Sasikumar, A., Ravi, L., Devarajan, M., Selvalakshmi, A., Almaktoom, A.T., Almazyad, A.S., Xiong, G. and Mohamed, A.W., 2024. Blockchain-assisted hierarchical attribute-based encryption scheme for secure information sharing in industrial internet of things. *IEEE Access*, 12, pp.12586-12601.
- [31] Sugier, J., 2024, June. Dedicated FPGA resources in improving power efficiency of implementations of BLAKE3 hash function. In *International Conference on Dependability of Computer Systems* (pp. 283-295). Cham: Springer Nature Switzerland.
- [32] Zhao, X., Zhang, G., Long, H.W. and Si, Y.W., 2024. Minimizing block incentive volatility through Verkle tree-based dynamic transaction storage. *Decision Support Systems*, 180, p.114180.
- [33] Christyono, B.B.A., Widjaja, M. and Wicaksana, A., 2021. Go-Ethereum for electronic voting system using clique as proof-of-authority. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 19(5), pp.1565-1572.