

Article

Optimizing Smart-Home Energy Systems Through Energy-Efficient Off-Chain Blockchain-Based Attribute-Based Access Control (ABAC): A Hybrid LightGBM Approach

Urooj Waheed ¹, Yusra Mansoor ^{1,*}, Najeeb Ur Rehman Malik ², Huma Jamshed ¹, Muhammad I. Masud ^{3,*}, Ahmed M. Nahhas ⁴, Mohammed Aman ⁵ and Touqeer Ahmed Jumani ⁶

¹ Department of Computer Science, DHA Suffa University, Karachi 75500, Pakistan; urooj.waheed@dsu.edu.pk (U.W.); huma.jamshed@dsu.edu.pk (H.J.)

² Department of Information Technology, DHA Suffa University, Karachi 75500, Pakistan; najeeb.malik@dsu.edu.pk

³ Department of Electrical Engineering, College of Engineering, University of Business and Technology, Jeddah 21361, Saudi Arabia

⁴ Department of Electrical Engineering, Faculty of Engineering and Architecture, Umm Al Qura University, Makkah 24382, Saudi Arabia; amnahhas@uqu.edu.sa

⁵ Department of Industrial Engineering, College of Engineering, University of Business and Technology, Jeddah 21361, Saudi Arabia; m.aman@ubt.edu.sa

⁶ Department of Electrical Engineering and Computer Science, College of Engineering, A'Sharqiyah University, Ibra 400, Oman; touqeer.ahmed@asu.edu.om

* Correspondence: yusra.mansoor@dsu.edu.pk (Y.M.); m.masud@ubt.edu.sa (M.I.M.)

Abstract

The widespread deployment of Internet of Things (IoT) technologies in smart-home energy systems has increased the demand for secure, context-aware, and energy-efficient access control (AC) mechanisms. Although blockchain-based AC provides immutability, auditability, and fine-grained policy enforcement, its dependence on on-chain decision-making introduces significant computational latency and energy overhead, limiting its suitability for resource-constrained IoT environments. This paper proposes Optimized Dynamic-Attribute-Based Access Control-IoT (ODABAC-IoT), a hybrid off-chain and decentralized ABAC framework that combines off-chain LightGBM inference with selective on-chain verification to reduce blockchain workload while preserving trust and transparency. This work focuses on improving the computational efficiency, latency, and energy consumption of blockchain-enabled AC within smart-home energy systems, rather than directly optimizing physical energy consumption. In the proposed framework, high-confidence access requests are evaluated off-chain, whereas uncertain requests are forwarded to smart contracts for final validation. This hybrid decision-making strategy reduces unnecessary blockchain transactions, lowers latency, and improves computational efficiency without compromising security. Experimental results demonstrate up to 65% reduction in blockchain transaction volume, 64% improvement in latency, and 65% reduction compared to on-chain ABAC and 50% compared to hybrid blockchain approaches. These gains correspond to a reduction in daily blockchain energy consumption from 10 kWh to 3.5 kWh in a representative household scenario. The results indicate that ODABAC-IoT improves scalability, energy efficiency of the digital control layer, and responsiveness in IoT-enabled smart home energy systems, offering an effective pathway toward energy-aware and secure AC in the digital infrastructure of smart home energy systems.



Received: 30 March 2026

Revised: 25 April 2026

Accepted: 6 May 2026

Published: 8 May 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article

distributed under the terms and

conditions of the [Creative Commons](#)

[Attribution \(CC BY\)](#) license.

Keywords: Internet of Things (IoT); smart-home energy systems; energy optimization; sustainability; blockchain; machine learning; hybrid systems; off-chain processing; smart-grid security; access control

1. Introduction

The increasing demand of the world towards more energy-efficient and sustainable technologies has enhanced the necessity of more complex control of electricity use and efficient management in interconnected surroundings. In smart-home and smart-grid scenarios, there is a wide range of IoT devices constantly communicating to balance loads, distributed generation, and overall energy efficiency. However, despite providing a high level of security and transparency, blockchain-based AC often results in both computational and energy overhead, which can undermine the efficiency benefits themselves. As a result, an increasing demand is building around lightweight and intelligent frameworks to ensure controlled and energy-aware operation of IoT-enabled energy systems.

IoT is becoming an increasing mainstay in industries such as healthcare, smart cities, and industrial automation. Nevertheless, as IoT Networks grow, concerns about AC, security, and scalability have also increased. Hasan et al. in [1] proposed a smart-contract-based AC framework addressing these challenges by utilizing blockchain for decentralized and secure policy enforcement. Waheed et al. [2] proposes a dynamic ABAC framework that integrated off-chain machine learning (ML) to reduce computational costs and improved system responsiveness by pre-evaluating access requests before they were processed on the blockchain, addressing the limitations of traditional Role-Based Access Control (RBAC), which is inflexible for fast-paced IoT environments with devices frequently joining and leaving networks. As a more context-sensitive solution, ABAC has potential but is not practically implemented due to complexity and insufficient decentralization [3].

Blockchain technology provides a decentralized way of AC, increasing the integrity and transparency of data sharing in IoT systems [4]. The use of smart contracts generated on a blockchain platform is beneficial as it provides automated and tamper-proof access decisions. However, using blockchain in the IoT presents issues in terms of performance and latency, notably with the availability of resources for low-powered devices [5].

In order to circumvent these issues, researchers have proposed hybrid models that pair blockchain with off-chain ML components. In a hybrid architecture, ML models can be employed to preprocess incoming access requests to filter out low-risk or high-confidence decisions and only submit ambiguous decisions to the blockchain, which reduces overhead [6]. LightGBM and Random Forest classifiers, for example, have shown good performance in improving the accuracy of access decisions and responsiveness of the system in a hybrid model architecture [7].

Recent studies have shown that ML can improve authentication and AC in resource-constrained IoT settings with good accuracy and lightweight performance [8]. Also, the use of distributed blockchain frameworks using ABAC can reduce single points of failure and improve scalability [9].

While these approaches improve AC efficiency, their impact on the energy consumption of the underlying blockchain infrastructure in smart-home energy systems remains insufficiently explored.

This paper proposes ODABAC-IoT, an off-chain and decentralized ABAC framework for IoT. ODABAC-IoT employs a preprocessing layer using LightGBM to assess access requests off-chain. Only those requests that are uncertain are forwarded to on-chain smart contracts for final validation. This represents a meaningful decrease in both the volume

of blockchain transactions and latency, while retaining the audit ability and transparency properties/benefits of the model. The proposed framework focuses on optimizing the computational and energy efficiency of the digital AC layer within smart-home energy systems, rather than directly modifying physical energy consumption patterns. This proposed framework attempts to address the performance, scalability, and security shortcomings of existing IoT AC solutions [10].

The contributions of this work are threefold:

1. Optimization of a hybrid blockchain architecture decision-making pipeline with integration of off-chain ML implication with on-chain verification.
2. Optimization of cost and performance through selective on-chain execution and periodic audit mechanisms.
3. Optimization of system efficiency through comprehensive evaluation demonstrating reductions in on-chain transaction volume, latency, and blockchain-related energy consumption with negligible loss in decision accuracy.

This paper is structured into the following: Section 2 carries out a detailed literature review of the related approaches, outlining current methodologies and highlighting the existing limitations thereof. Section 3 elaborates the presented ODABAC-IoT framework, including its architectural elements and design principles. Section 4 shows how the framework can be applied to the example of a smart-home energy-management process. Section 5 describes the experimental results and provides a detailed discussion, including transaction-volume reduction, latency reduction, storage optimization, blockchain-related energy savings, compliance with energy standards, and a general evaluation of the framework performance.

2. Related Work

2.1. Blockchain-Based Access Control in the IoT

The rapid growth of the Internet of Things (IoT) means AC now faces considerable challenges related to device diversity, dynamic network topologies, and the need for more secure and real-time decision making [11]. Traditional AC models such as RBAC and ABAC have made their way into IoT ecosystems. RBAC enforces AC through roles, meaning IP devices are assigned roles, and permissions are based upon these roles [12]. However, ABAC policies are enforced using various attributes, including user, device, and environmental attributes, which allow for a more dynamic approach, and to varying degrees, support finer-grain and context-aware AC decisions [13]. Because it is less prescriptive in such respects, it is seen as a suitable model in a number of IoT applications compared to RBAC [14].

IoT devices are often lightweight and work under stringent constraints in terms of resources and power, which usually lack enough computational or storage capabilities to perform complicated tasks related to AC on their own. As a result of this, conventional AC models and their cryptographic analog called ABE are often built on the use of centralized management systems that allow for the central evaluation of policies, and for the central encryption and decryption of data, as well as allowing for central decision-making about who can access which data [15]. This reliance on and management by a central system creates a number of serious problems including single-point failures and greater exposure to manipulation, and limits the ability to grow in size or scale [16]. Although ABE is able to give assurance to a data owner that only those individuals with credentials that meet or exceed the qualifications specified in the owner's policy for that data are able to decrypt it, the cost of executing ABE algorithms on an IoT device is extremely high [17]. In addition, although ABE is specifically designed to protect the confidentiality of stored data, it is not designed to provide real-time AC or dynamic service authorizations in the face of changing circumstances [18].

To address these constraints, newer frameworks propose combining lightweight intrusion detection, edge computing, and blockchain to decentralize and optimize AC for real-time, scalable, and secure IoT deployment [19].

Blockchain technology has been seen as a solution to data AC issues in the IoT because of its decentralization, immutability, and consensus mechanisms for verification [20]. In RBAC, blockchain can provide secure storage of roles and permissions in the distributed ledger, while smart contracts manage automated role assignment, verification, and tracking of misbehavior [21]. With ABAC, the inherent intelligence of smart contracts can provide real-time evaluation of policies and matching dynamic sets of attributes, allowing for real-time and adaptive policy enforcement in a dynamic context and permitting differentiated access to public and private data of IoT devices [22]. Dynamic ABAC implementations can also utilize immutable logs of all entries (unique identification) into distributed services controlled by the entity, decentralized attribute control, and dynamic or context-aware attribute verification and updates from whichever trusted or untrusted third parties (cross-organizational) are required or authorized by the entity [23]. These characteristics enable scalable, interoperable, and fully accountable permissions, rights management, and transparent AC in the IoT across once untrusted third parties or across organizations [24].

2.2. Dynamic ABAC Models and Computational Challenges

Blockchain-enabled ABAC has been greatly studied for secure and decentralized policy implementation in IoT systems; however, it has many limitations in practical implementations because of high computational overheads in resource-pairing-constrained environments.

As described by the recent literature, the most significant challenges of blockchain-enabled ABAC models are the costly rate of computation arising from multiple evaluations of attributes [25], the high cost of smart-contract execution within blockchain platforms [26], and consensus mechanisms that contribute to high latency [27]. In the case of ABE (such as bilinear-pairing-based ABE), the rate of computation also imposes heavy processing loads on small and lightweight IoT devices [28]. Additionally, recording all access data in granular, on-chain logs increases ledger size and creates inefficiencies in operational data storage, scaling, and transaction costs [29].

Fine-grained ABAC policies on blockchain yield better decision performance, but they incur high fees and longer response times in public blockchain scenarios [30]. In addition, ref. [31] reckons that ABAC's cryptographic requirements impose substantial issues in low-power IoT devices and do not yield sufficient and timely AC. To overcome these challenges, there has been a wave of new research into hybrid and off-chain computing architectures, which can reduce on-chain workloads while still being transparent and auditable [32,33]. However, the general research issue is to introduce low-latency, energy-efficient, and scaling of blockchain-based access management systems using ABAC in the IoT setting [34].

2.3. AI Driven Blockchain Optimization

Recent research has looked into the ways in which Artificial Intelligence (AI) can be integrated with a blockchain to improve scalability, security, and operational efficiencies [35]. The authors of [36,37] thoroughly reviewed AI-based blockchain approaches, providing coverage of AI's usefulness in optimizing consensus mechanisms, identifying vulnerabilities in smart contracts, and enabling privacy-preserving computations. The survey authors reported that ML and DL, AI approaches, were beneficial to improve consensus latency, predict malicious behavior, and facilitate improved contract execution [38]. However, other thematically similar AI-enabled AC mechanisms geared toward IoT (Internet of Things)

systems have not been explored in detail, despite researchers mainly focusing on improving only the foundational operational efficiencies of blockchain [39].

Building on these innovations, Anand et al. in [40] proposed an AI and blockchain-based AC model that included biometric authentication, anomaly detection, and contextual analysis to facilitate dynamic management of ABAC policies. In that model, blockchain was utilized to conduct trustworthy, decentralized storage of control policies. Using AI, the model could process user and device behavior to continually update access rules for each user and their associated devices. This yielded an overall enhancement with the ML aspect for the overall identity management in characteristics such as consent, also supporting the development of privacy-preserving federated learning and enhanced security by increased resilience against evolving attacks [41,42]. Similarly, the works of [43] and others demonstrate how AI-based inference methods can reduce oversight over manual policies and speed incident response, adapt AC decisions substantially quicker, more smoothly, and efficiently respond to changing security contexts.

While these advantages exist, substantial computational challenges still exist. The joint load of AI inference and smart-contract execution may be excessive in resource-limited IoT devices and limit the service latency and transaction costs in the case of a public blockchain. The two studies above emphasized that future research should focus on balancing lightweight AI models, off-chain processing, and edge computing based on the security strength and performance efficiency of AI blockchain-enabled IoT AC systems.

2.4. Lightweight Blockchain Frameworks with Off-Chain Intelligence

Off-chain computation has emerged as an effective strategy for addressing the scalability and performance limitations of blockchain-based systems, particularly in latency-sensitive and data-intensive IoT applications. By executing computational tasks outside the blockchain and only committing essential proofs or outcomes on-chain, this approach preserves decentralization and verifiability while significantly reducing storage, transaction costs, and consensus latency [44].

Yang et al. [45] identified off-chain technologies as comprising privacy mechanisms, intelligent networking, edge computing, and transaction scheduling, which extend to adopting advanced AI approaches, such as federated learning and deep reinforcement learning, to optimize routing and resource assignment and to assess trust without adding overhead to the blockchain layer. The AI models can be trained and executed in an off-chain manner, but the correctness, integrity, accountability, and auditability are taken care of on-chain, along with the secure enforcement of policy implications.

In a hierarchy of blockchain presented by [46] for smart cities, the presented system solution consists of the management of resource and location of diverse IoT nodes using both public and private chains, leveraging lightweight cryptographic methods and utilizing ML-based inconsistency detection. The main limitation of the proposed system is the lack of an execution environment for off-chain real-time detection, and it misses the opportunity to better incorporate the descriptive and prescriptive power of AI [47,48].

Similarly, Hakilo Sabit in [49] proposed a smart-home application leveraging off-chain ML inference to only trigger aspects of actions captured on the blockchain in the interest of minimizing the frequency of on-chain transaction processing and thus reducing transaction costs. They also solely considered the actions as events to be automated, refraining from studying the impact on dynamic AC or a more modular and unique AC based on policies. Emerging directions include integrating federated learning with off-chain storage (e.g., IPFS) for privacy-preserving model training in distributed IoT systems [50,51], and reinforcement learning-based routing in Payment Channel Networks, which improves throughput and privacy [52]. In IoT AC, such off-chain intelligence can support real-time

anomaly detection and policy evaluation at the edge, with only final authorization results recorded on-chain for immutability and audit.

Despite these benefits, off-chain AI systems still face challenges, such as high computational demands, trust gaps between off-chain and on-chain components, and ensuring transparent, verifiable decision-making. Future research is expected to focus on hybrid designs combining lightweight edge AI, cryptographic proofs of computation, and selective on-chain validation to balance security, scalability, and performance in blockchain-enabled IoT AC. As shown in Table 1, the various AC techniques in IoT systems are summarized, highlighting their key features, limitations, and applications.

Table 1. Summary of access control techniques in IoT studies.

Reference	Study Focus	Access Control Approach	Key Features	Limitations
[12]	RBAC-based secure data processing in IoT	RBAC	Secure and efficient framework for IoT data processing	Limited adaptability for dynamic environments
[14]	Context-aware access via fog computing	Context-aware AC	Fog-enabled dynamic access policy adaptation	Complexity in context integration
[16]	Zero-trust access for 6G IoT environments	Zero trust + blockchain + inner product encryption	Designed for 6G-enabled IoT networks	High computation overhead and encryption complexity
[17]	Healthcare access with ABE	Attribute-Based Encryption (ABE)	Secure-cloud-based system with fine-grained control	Tailored for healthcare, limited generalizability
[18]	Leakage-resilient control for IoT outsourcing	Leakage-resilient AC	Resists data leakage during outsourced storage	Focuses more on leakage than real-time access
[19]	AI-driven adaptive threat detection in IoT	AI-integrated AC	Real-time adaptive security under constraints	Lacks blockchain integration
[20]	Decentralized authentication in IoT access	Hybrid access control with decentralized auth	Emphasizes authentication and layered access	Review-based, lacks implementation insights
[21]	Policy-based control for health data	Policy-based AC with blockchain	Unauthorized access prevention in EHR systems	Application-specific (healthcare), scalability untested
[22]	Boundary-access enforcement in IoT	Fine-grained trusted control	Enhanced boundary control, fine-grained policies	Boundary specific, not generalized
[23]	DID-based identity for IoT devices	Decentralized Identity Management (DID)	Blockchain-based identity for IoT devices	Relies on IOTA, may limit interoperability
[24]	Blockchain-enabled IoT data marketplaces	Use-case-oriented blockchain integration	Market-driven IoT data control	Use-case-centric, lacks generalizable access model
[25]	Scalability in blockchain IoT access	Performance-oriented blockchain access	Analyzes performance in varied IoT settings	Focuses on performance, not access policy design
[26]	Policy-controlled smart contracts in IoT	Policy-based smart contracts	IoT privacy via policy managed smart contracts	Privacy heavy, lacks performance benchmarking
[27]	Smart-city access control using 6G + blockchain	Blockchain IoT in smart cities	6G integration, secure urban IoT	Generalized, not focused on specific control mechanisms
[28]	Ethereum smart contracts for IoT access	Ethereum-based ABAC	RBAC and Ethereum smart contracts' integration	Requires Ethereum network, high cost

Table 1. *Cont.*

Reference	Study Focus	Access Control Approach	Key Features	Limitations
[29]	Hybrid storage in blockchain traceability	On-chain + off-chain access strategy	Enhanced scalability with hybrid storage	Focused on traceability, not full access control
[30]	ABAC for blockchain donation systems	ABAC	Donor centric access transparency	Niche application, limited IoT focus
[31]	Fine-grained control in blockchain IoT	Blockchain-based fine-grained access	Secure storage and access granularity	General IoT focus, lacks trust management
[32]	Distributed access in Industrial IoT	Distributed-blockchain AC	Access for IIoT through blockchain enforcement	Tailored to industrial settings
[33]	Supply-chain access visibility using blockchain	Blockchain-enabled supply-chain control	Track and access visibility	Limited to logistics-based IoT
[35]	Convergence of blockchain IoT AI for access	Converged-access frameworks	Multitech synergy for access and control	Early stage, lacks prototype
[37]	Smart-contract administration in virtual settings	Blockchain-based contract management	Optimized contract handling	Focused more on admin than access policies
[38]	Digital-twin security with IoT blockchain	Blockchain-based digital-twin security	Secure twin representation and monitoring	Conceptual, lacks deployment examples
[39]	Governance framework for blockchain IoT AI	Conceptual governance framework	Access governance across AI, IoT, blockchain	Conceptual, not technical
[40]	Blockchain with TL for secure dissemination	Blockchain-based dissemination control	Transfer learning + blockchain for data spread	Dissemination focused, less on access policy

The evolution of access-control mechanisms in IoT systems has progressed from centralized RBAC models to decentralized ABAC frameworks and more recently, to hybrid blockchain–ML architectures that aim to balance security and efficiency.

Compared to existing hybrid blockchain-based access control frameworks, ODABAC-IoT introduces a more efficient decision-making pipeline by integrating lightweight LightGBM inference with selective on-chain verification. While prior works such as [1,2] focus on fully on-chain or partially optimized blockchain execution, they still incur significant computational overhead due to frequent smart-contract evaluations. In contrast, the proposed framework reduces unnecessary on-chain interactions by filtering high-confidence decisions off-chain, thereby achieving lower latency and energy consumption. Furthermore, unlike conventional approaches, ODABAC-IoT incorporates periodic auditing and trust recalibration mechanisms to maintain security guarantees while improving efficiency. These features collectively position the proposed framework as a scalable and energy-aware alternative to existing state-of-the-art solutions.

3. Methodology

The proposed ODABAC-IoT framework is an optimized hybrid blockchain-enabled AC mechanism for smart-home energy systems. The methodology is built directly on a hybrid blockchain-based dynamic ABAC framework for smart energy environments. It combines Hyperledger Fabric and Hyperledger Besu to achieve decentralized, tamper-proof, and context-sensitive AC while ensuring strong security and operational efficiency and to overcome the computational overhead and scalability challenges arising from the high frequency of on-chain access evaluations. This model also introduces an off-chain Light Gradient-Boosted Machine (LightGBM)-driven pre-evaluation layer. LightGBM is a gradient-

boosting decision tree framework specifically optimized for high-speed inference on tabular data, which aligns with the structured and context-rich nature of IoT AC attributes. Its ability to handle categorical, numerical, and temporal features efficiently makes it well suited for processing heterogeneous IoT request data in real time. The methodology focuses on optimizing the computational efficiency, latency, and blockchain-related energy consumption of the digital access control layer, rather than directly modifying physical energy consumption in the smart home. In the proposed framework, high-confidence decisions are executed off-chain using the lightweight LightGBM classifier, while low-confidence requests are enforced for full blockchain-based ABAC implementation. The low latency implication of LightGBM, along with a small memory footprint, makes it a suitable choice for deployment on edge devices and ensures scalability in resource-constrained IoT environments.

Trust is preserved using cryptographic signing, and audit logs ensure integrity and traceability, enabling the system to attain substantial cost and latency reductions without compromising immutability or transparency. Unless otherwise stated, all performance and energy-related results refer to the computational and blockchain infrastructure layer.

The architecture of the proposed ODABAC-IoT framework is shown in Figure 1 and consists of three layers: the IoT Device Layer, the Off-Channel ML Processing Layer, and the On-Channel Blockchain Layer. The first layer comprises smart meters, appliances, and energy management controllers that generate access requests along with contextual attributes. The Off-chain ML Processing Layer, which serves as the second layer, includes LightGBM as a prediction engine, hosted on edge nodes, to enable high-speed evaluation of access requests. Finally, the third on-chain blockchain layer utilizes the Hyperledger Fabric network to host smart contracts for policy enforcement, secure logging, and continuous trust recalibration. The solid line represents a compulsory interaction, whereas the dotted line represents a conditional interaction between the components. The white lines represent the optimization in the framework proposed in [2] study, whereas the black lines represent the adoption of the framework described by Waheed et al. in [2].

In the proposed architecture, in keeping with the zero-trust model, there is no trust implicit at any point in the system; every request for access is verified contextually via LightGBM inference and then validated again by blockchain-enabled ABAC policies.

This makes it possible to continuously adjust trust according to user behavior, thus applying the principle of least privilege. The use of off-chain ML improves decision-making speed without compromising on validation, whereas on-chain auditing ensures nothing escapes cryptographic scrutiny. Latency refers to the period of time taken between the creation of an access request and the ultimate decision made to grant permission.

3.1. IoT Device Layer

This includes smart-home devices and energy management components that continuously interact with the Home Energy Management System (HEMS) and external utility services and serve as the data generation and request initiation point inside the ODABAC-IoT framework. To be specific, this layer comprises smart meters for household consumption and tariff status reporting, HVAC systems (air conditioners and water heaters) for controllable load management, renewable energy units (solar inverters), and energy storage systems that manage charging and discharging cycles. Also, EV chargers, demand response controllers, and environmental sensors for monitoring temperature, occupancy, voltage, and frequency. Over time, these devices, generate requests for access like “turn on the EV charger for 2 h,” “switch air conditioner on at 5 p.m.,” etc., so they are categorized into static (device ID, owner ID, manufacturer), dynamic (energy usage, device state, battery SOC, current tariff), and contextual (location, time of request, occupancy, peak/off peak periods) classes.

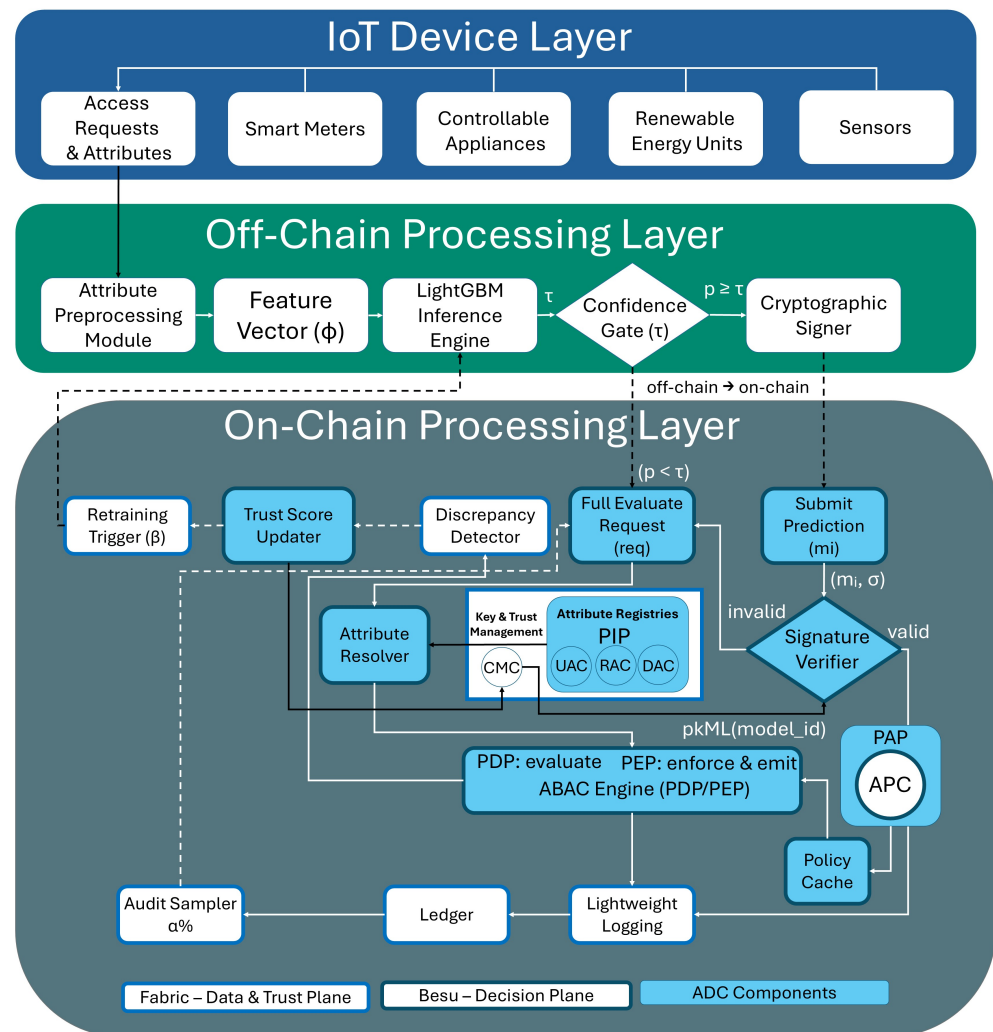


Figure 1. ODABAC-IoT hybrid architecture with off-chain ML and on-chain verification.

The AC pipeline's initial stage is formed by this layer, which guarantees that every single piece of operational and contextual data is collected. This in turn allows for aware and fine-grained context both for LightGBM off-chain inference and ABAC policy execution. The collected attributes are assumed to represent typical smart-home operational conditions and are used as inputs for both off-chain ML inference and on-chain policy evaluation.

3.2. Off-Chain Processing

All access requests are preevaluated off-chain before they experience blockchain processing costs. The off-chain processing pipeline in Figure 2 serves as the first level of intelligence in the ODABAC-IoT architecture, transforming raw IoT attributes into clean, standardized, and contextually enriched feature vectors, and leveraging LightGBM to make fast access predictions with high confidence. It balances efficiency and trust, and the model is trained against audited ABAC decisions, i.e., reducing blockchain execution overheads while still maintaining security and flexibility as indicated by conditional interaction represented by dotted line with in the framework. Thus, it reduces blockchain execution costs and preserves security and adaptability. This tiered architecture ensures on-chain reliable AC outcomes optimizing both system performance and policy robustness in dynamic IoT environments. The concept of an off-chain processing phase assumes that an access request's attributes and distribution will match standard models that are used in typical smart-home application systems; therefore, obtaining a generalized solution to this

problem becomes feasible. The training set includes simulated access requests based on ABAC on a blockchain and IOT access request simulations based on normal and abnormal access patterns.

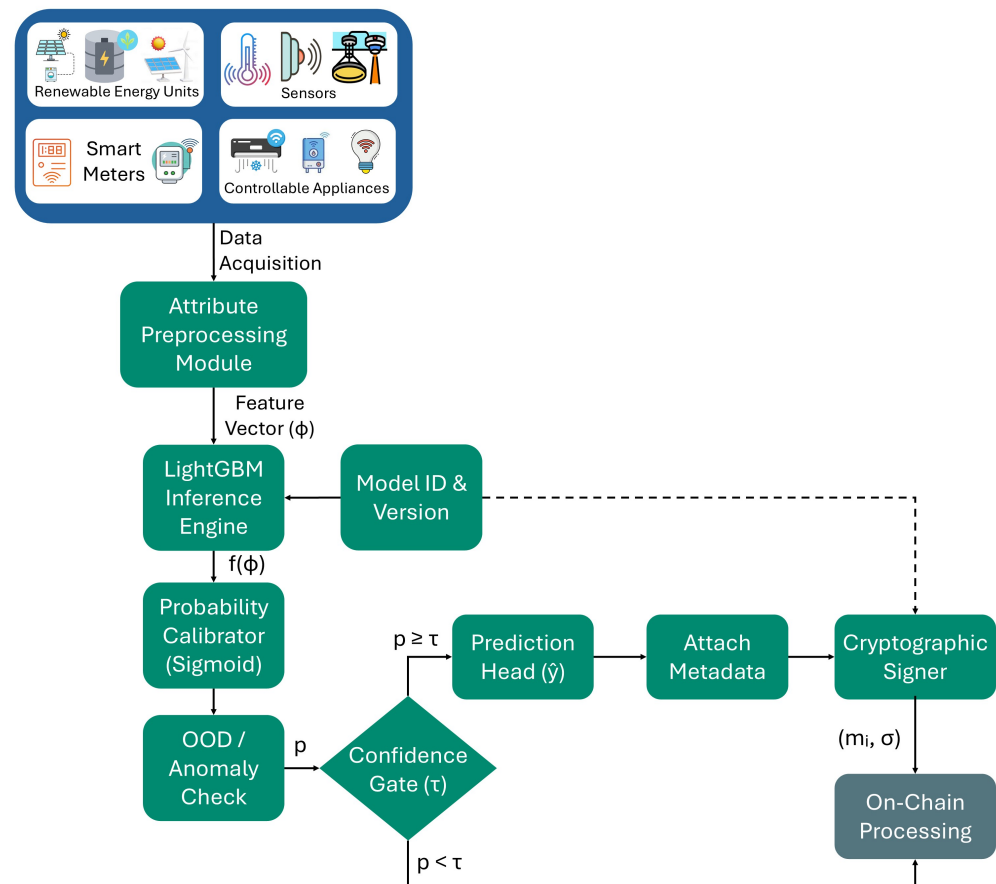


Figure 2. Computation flow of off-chain processing in ODABAC-IoT framework.

3.2.1. Attribute Acquisition and Preprocessing

Each AC decision in IoT environments can be represented as a collection of attributes from four domains:

- **User attributes:** identity, role, authentication level, past behavior history.
- **Device attributes:** type of device (e.g., air conditioner, EV charger), power rating, criticality level.
- **Request attributes:** action requested (turn on/off, limit usage), requested duration, resource demand.
- **Contextual attributes:** time of day, tariff period (peak/off-peak), household occupancy, grid load state, renewable availability.

It is assumed that all attributes are time-synchronized at the moment of request generation and that missing values occur randomly rather than systematically. The data from the IoT devices are collected through secure channels and passed to the preprocessing module, as shown in Figure 3. These collected data are first processed for missing values by median imputation to ensure data completeness while maintaining distributional integrity. Next, robust anomaly scoring using the robust z score is applied to identify and mitigate outliers arising from sensor noise or malicious injections. The cleaned data then undergo standardization of numerical attributes, transforming them to zero mean and unit variance for stable learning.

For energy-related readings, device-aware normalization is employed to scale features relative to each device's operational range. Discrete variables are handled through categorical encoding, converting symbolic attributes into machine-readable form. Temporal and contextual variables are processed with cyclical encoding, preserving natural periodicity such as daily or weekly cycles. Finally, all transformed features are combined through feature assembly, producing a unified vector (x_i) suitable for input to the ML prediction engine. The raw attribute vector for a request at time t_i can be expressed using Equation (1), where u_i , v_i , r_i , and c_i denote user, device, request and contextual attributes respectively.

$$x_i = [u_i, v_i, r_i, c_i] \in \mathbb{R}^d \quad (1)$$

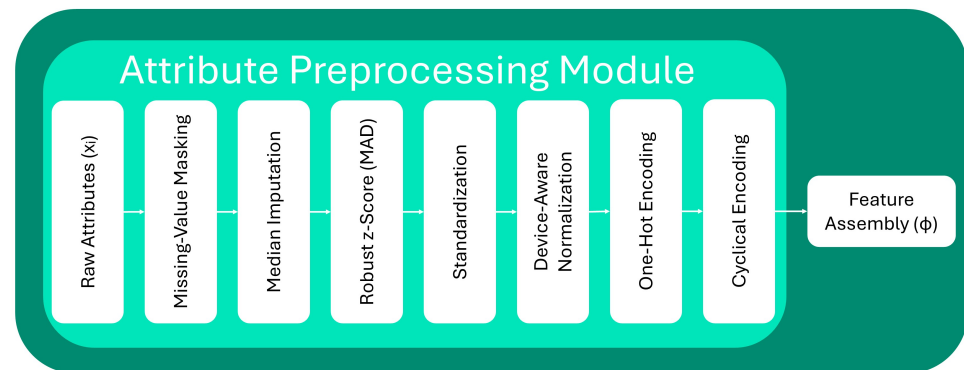


Figure 3. Attribute preprocessing module for IoT access requests.

Missing-Value Masking and Median Imputation

For each feature j , a binary mask $m_i \in [0, 1]^d$ indicates whether the value is observed ($m_{ij} = 1$) or missing ($m_{ij} = 0$). Missing values are replaced using median-based imputation. For each feature i , let med_j be the median computed on the training set as presented in Equation (2).

$$\tilde{x}_{ij} = m_{i,j} \cdot x_{i,j} + (1 - m_{i,j}) \cdot med_j \quad (2)$$

This assumes that missing values do not carry semantic meaning and can be approximated using median-based imputation without introducing significant bias.

Robust Anomaly Scoring (Robust z-Score)

The robust z-score is used to flag anomalous values, where the median med_j provides a robust estimate of central tendency, and the median absolute deviation MAD_j captures the spread of each attribute as shown in Equation (3).

$$MAD_j = median_k(|x_{kj} - med_j|) \quad (3)$$

The robust z-score ensures that occasional noisy readings (e.g., sudden spikes in smart-meter values or abnormal temperature readings from sensors) do not distort the preprocessing pipeline. This allows the system to reliably detect and handle anomalous attribute values before they are passed to the LightGBM classifier or the blockchain-based ABAC engine, thereby reducing false denials and improving model robustness in dynamic IoT environments. Therefore, the normalized distance of a data point $x_{i,j}$ from the median of that attribute, scaled by the typical spread (MAD) is shown in Equation (4). Thus, r_{ij} is basically an anomaly score per attribute per request.

$$a = \frac{\tilde{x}_{ij} - med_j}{1.4826 \cdot MAD_j} \quad (4)$$

where

- x_{ij} = value of attribute j for data instance i
- med_j = median of attribute j
- MAD_j = median absolute deviation, a robust measure of spread.
- The constant 1.4826 ensures consistency with standard deviation under normal distribution assumptions.

Standardization of Numerical Attributes

Numerical attributes excluding power/energy features that will undergo device-aware scaling are standardized to ensure comparability across devices with different measurement scales. Standardization transforms each attribute x_{ij} into $z_{i,j}$ as shown in Equation (5)

$$z_{i,j} = \frac{x_{i,j} \cdot \mu_j}{\sigma_j} \quad (5)$$

where μ_j and σ_j denote the mean and standard deviation of the attribute. This yields zero mean, unit variance features so attributes with larger raw ranges (e.g., latency vs. temperature) do not dominate the model.

Device-Aware Normalization for Power/Energy Features

For power and energy consumption, device-aware normalization is applied. Each device's readings are scaled relative to its own operating range or rated capacity, so that both low-power devices (e.g., bulbs) and high-power devices (e.g., EV chargers) contribute comparably to the classifier. For instance, the min and max consumption of device d is known then the normalized power/energy value of feature j for request i from device d can be expressed as shown in Equation (6).

$$p_{i,j}^{(d)} = \frac{x_{ij} - \min_d}{\max_d - \min_d} \quad (6)$$

Similarly, if the rated capacity of device d is known, then normalized power/energy value of feature j for request i from device d can be expressed as shown in Equation (7).

$$p_{i,j}^{(d)} = \frac{x_{ij}}{\text{rated} \cdot d_d} \quad (7)$$

This assumes that device operational ranges and rated capacities are known and remain stable over time.

Categorical Encoding

Categorical attributes are transformed via one-hot encoding. Formally, if feature j takes value $x_{ij} = k_k$ (the K^{th} category), the encoding function maps it to the one-hot vector as shown in Equation (8)

$$\phi_j(x_{ij}) = \mathbf{e}_k, \quad x_{ij} = \kappa_k. \quad (8)$$

where $\mathbf{e}_k$ is a K -dimensional unit vector with 1 at the k^{th} position and 0 elsewhere.

Cyclical Encoding for Temporal/Contextual Variables

Temporal/contextual variables such as time of day and day of week are encoded as cyclical features as expressed in Equation (9)

$$t_i^{\sin} = \sin\left(\frac{2\pi \cdot h_i}{24}\right), \quad t_i^{\cos} = \cos\left(\frac{2\pi \cdot h_i}{24}\right). \quad (9)$$

where t_i indicates time of day or day of week.

This unified feature representation ensures that heterogeneous IoT attributes are transformed into a consistent numerical space suitable for ML inference.

Feature Assembly

The complete preprocessing pipeline yields a model-ready feature vector as represented in Equation (10)

$$\phi_i = [\mathbf{z}_i \parallel \mathbf{p}_i^{(d)} \parallel \mathbf{o}_i \parallel \mathbf{t}_i \parallel \mathbf{a}_i] \in \mathbb{R}^{d'}. \quad (10)$$

where

1. $\mathbf{Z}_i$ is a standardized numeric;
2. $\mathbf{p}_i^{(d)}$ is the device-aware power/energy;
3. $\mathbf{o}_i$ represents the one-hot encoding;
4. $\mathbf{t}_i$ is the cyclical time/context;
5. $\mathbf{a}_i$ are anomaly flags.

This formalized preprocessing ensures that attributes collected from IoT devices are clean, normalized, and contextually enriched before being passed to the off-chain inference engine and ABAC policy evaluator.

3.2.2. Machine Learning Prediction Engine

The ML Prediction Engine is an off-chain module responsible for making fast and reliable AC predictions. Algorithm 1 and Figure 4 illustrates the off-chain ML Prediction Engine workflow within the ODABAC-IoT framework with conditional interaction in the form of dotted line, feature importance and explainable AI. Attribute data from IoT devices, contextual features, and trust recalibration history are preprocessed into feature vectors ϕ_i . These vectors are passed to the LightGBM model, which iteratively combines shallow decision trees to produce a predicted access decision with calibrated confidence $p_i \in [0, 1]$ as shown in Equation (11).

$$y_i \in \{\text{Allow}, \text{Deny}\} \quad (11)$$

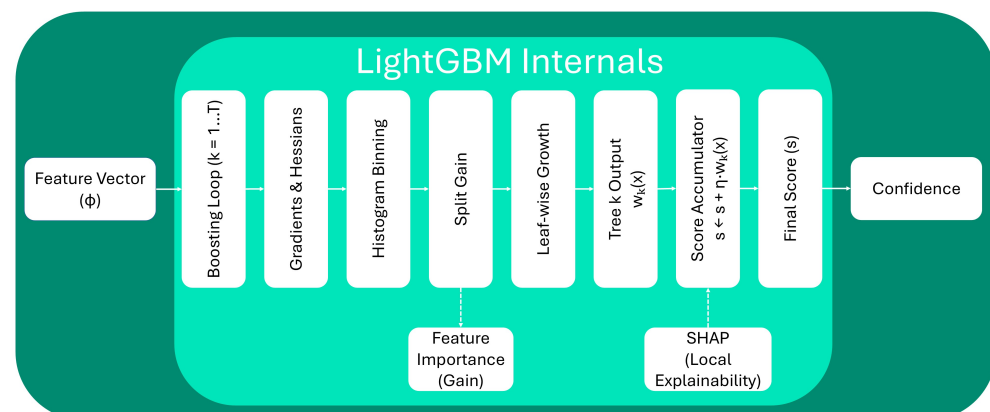


Figure 4. Machine learning prediction engine pipeline in ODABAC-IoT.

The prediction model is implemented using Gradient-Boosted Decision Trees (GBDTs), an ensemble learning approach that builds a strong classifier by combining the outputs of many weak learners, typically shallow decision trees. Unlike a single decision tree, which often suffers from overfitting or limited generalization, GBDTs improve performance by training trees sequentially, where each new tree is optimized to correct the residual errors of the previous ensemble. The training dataset was extracted from CouchDB off-chain storage, which contains historical attribute decision pairs along with trust recalibration history. This ensured that the model was trained on realistic access patterns and could generalize well to future requests. Formally, let the cleaned training dataset be as shown in Equation (12):

$$\mathcal{D} = \{ (\phi_i, y_i) \}_{i=1}^N \quad (12)$$

where ϕ_i are the feature vector and $y_i \in [0, 1]$ are the binary labels.

The off-chain database (CouchDB) that provides the training dataset for the LightGBM model was based on historical access-control logs, containing attribute–decision pairs produced as a result of previous ABAC checks. To preserve privacy, user identifications were not utilized in any way; only anonymous attributes were considered as contextual and behavioral features. Additionally, using off-chain technology for training purposes helped protect personal information by keeping the raw sensitive data away from the blockchain.

An 80:20 split was used to create both an 80% (training subset) and a 20% (validation subset) subset of the total dataset in order to allow for an assessment of the generalizability of the final classification performance. The model built using LightGBM on the training subset and its performance were assessed using a validation (testing) subset. The classification performance of each method was evaluated using standard classification metrics of accuracy, precision, recall, and F1-score across each model and subset, and the models' robustness (i.e., their ability to prevent overfitting) were further evaluated through the use of k-fold cross-validation (where $k = 5$) to test the model with 5 different (random) partitions of the data.

The model aims to minimize a differentiable loss function. The classifier is trained by minimizing a loss function as shown in Equation (13), where y denotes the ground-truth access decision obtained from blockchain ABAC enforcement, ϕ represents the preprocessed attribute vector, and $f(\phi)$ is the predicted decision score from the LightGBM model.

$$L(y, f(\phi)) \quad (13)$$

The LightGBM training process begins with an initial constant prediction equal to the log odds of the positive class, i.e., the ratio of Allow vs. Deny outcomes in the blockchain-verified dataset as shown in Equation (14), where p is the proportion of “Allow” decisions in the training data.

$$F_0 = \log\left(\frac{p}{1-p}\right), \quad (14)$$

Training proceeds iteratively, where in each round residuals are computed as the difference between ground-truth blockchain decisions and the model's predictions. For instance, at each round m it measures the difference between the blockchain's decision u_i and the model's current prediction $f_{m1}(\phi_i)$ as presented in Equation (15)

$$r_{im} = u_i - f_{m1}(\phi_i) \quad (15)$$

A shallow regression tree is trained on the pairs (ϕ_i, r_{im}) to predict these residuals. This tree learns where the current model is making errors. The outputs of the new tree $h_m(\phi)$ are added to the model, scaled by a learning rate η as shown in Equation (16). This ensures gradual corrections and prevents overfitting.

$$f_m(\phi) = f_{m1}(\phi) + \eta \cdot h_m(\phi) \quad (16)$$

After M boosting rounds, the final LightGBM model is expressed in Equation (17)

$$f_M(\phi) = F_0 + \sum_{m=1}^M \eta \cdot h_m(\phi) \quad (17)$$

where $f_M(\phi)$ outputs a logit score. A sigmoid transformation converts it into the probability of an Allow decision as represented in Equation (18)

$$P(\text{Allow} \mid \phi) = \sigma(f_M(\phi)) = \frac{1}{1 + e^{-f_M(\phi)}} \quad (18)$$

The predicted probabilities are assumed to be well-calibrated outputs of the LightGBM model; however, no explicit calibration method is applied, which may introduce uncertainty in confidence estimation.

If the confidence score exceeds a predefined threshold τ , the decision is directly accepted and logged on-chain to minimize energy consumption. Otherwise, low-confidence cases are deferred to a full on-chain ABAC evaluation for stricter verification as shown in Equation (19).

$$y_i = \begin{cases} \text{Allow,} & \text{if } p_i \geq \tau, \\ \text{Deny,} & \text{if } p_i < \tau. \end{cases} \quad (19)$$

The confidence threshold τ is analytically set (e.g., $\tau = 0.9$) and controls the trade-off between off-chain efficiency and on-chain security verification.

Algorithm 1 Training and inference of ODABAC-IoT classifier

Input:

- Training dataset $\mathcal{D} = \{(\phi_i, u_i)\}_{i=1}^N$, where ϕ_i are preprocessed attribute vectors and u_i are blockchain ABAC labels;
- Learning rate η ; number of boosting rounds M .

Output:

- Trained LightGBM classifier $f_M(\phi)$.

Training Phase

- 1: *Initialization*: Compute initial model score (logit baseline):

$$F_0 = \log\left(\frac{p}{1-p}\right), \quad p = \frac{1}{N} \sum_{i=1}^N u_i$$

Set $f_0(\phi) = F_0$.

- 2: *Boosting Rounds* (for $m = 1$ to M):

- (a) Pseudo residuals for each sample:

$$r_{im} = u_i \sigma(f_{m1}(\phi_i))$$

- (b) Fit a regression tree $h_m(\phi)$ to map $\phi_i \mapsto r_{im}$.

- (c) Update the model:

$$f_m(\phi) = f_{m1}(\phi) + \eta \cdot h_m(\phi)$$

- 3: *Return final model*:

$$f_M(\phi) = F_0 + \sum_{m=1}^M \eta \cdot h_m(\phi)$$

Inference Phase (Access Request Evaluation)

- 1: For an incoming IoT request with raw attributes $\mathbf{z}$, preprocess:

$$\phi = \Phi(\mathbf{z})$$

- 2: Compute logit score:

$$\hat{y} = f_M(\phi)$$

- 3: Convert to probability:

$$P(\text{Allow} \mid \phi) = \sigma(\hat{y}) = \frac{1}{1 + e^{-\hat{y}}}$$

- 4: Apply confidence threshold τ :

If $P \geq \tau$: decision = ALLOW, cryptographically signed, logged on-chain.

If $P < \tau$: escalate to full blockchain ABAC evaluation.

3.2.3. Cryptographic Result Signing

Adversarial attack resilience is a key element in the design of the suggested architecture. Being based on off-chain contextual information, the LightGBM-based ML algorithm can suffer from adversarial attacks, since the context itself might be manipulated by malicious actors. ODABAC-IoT addresses these challenges using multiple protective measures. First, all ML results with a high degree of confidence are signed cryptographically and checked via on-chain validation to make sure that no altered predictions could pass unnoticed. Second, any predictions with a low-confidence level are automatically forwarded for full-scale ABAC on-chain processing. Finally, on-chain model retraining on regular basis would allow spotting adversarial patterns. The combination of these techniques increases the resistance of the proposed architecture to adversarial attacks.

FFDI (false-data injection) attacks are a possible risk in which bad people can affect how ML is able to infer results by inserting their own data into the input attributes and then using these new “faked” attributes to change how ML infers results based on those altered attribute values. The proposed architecture provides multiple security measures to mitigate FDI attacks through each layer of the architecture. For example, there are several robust preprocessing methods, such as anomaly detection (for example, using the robust z-score mechanism), to identify abnormal or invalid values in the input attributes prior to performing inference. In addition, all low-confidence or suspicious predictions are redirected automatically to a complete on-chain ABAC evaluation to ensure that bad data do not get past the enforced policies. Periodic auditing and trust recalibration further enhance the system’s ability to detect and respond to inconsistencies caused by malicious data injection.

To preserve integrity and authenticity, every high-confidence prediction generated by the off-chain ML Prediction Engine is digitally signed before being forwarded on-chain as illustrated in Figure 5. It produces a decision tuple as shown in Equation (20)

$$m_i = (\hat{y}_i, p_i, t_i) \quad (20)$$

where y^i is the predicted decision, p_i is the confidence score, and t_i is the timestamp. The ML node uses its private key sk_{ML} to generate a digital signature as shown in Equation (21)

$$\sigma_i = \text{Sign}_{sk_{ML}}(H(m_i)) \quad (21)$$

where $H(\cdot)$ is a cryptographic hash function, and σ_i is the digital signature bound to the result tuple. On-chain, the blockchain nodes verify the signature using the ML node’s public key pk_{ML} as shown in Equation (22)

$$\text{Verify}_{pk_{ML}}(H(m_i), \sigma_i) \stackrel{?}{=} \text{True} \quad (22)$$

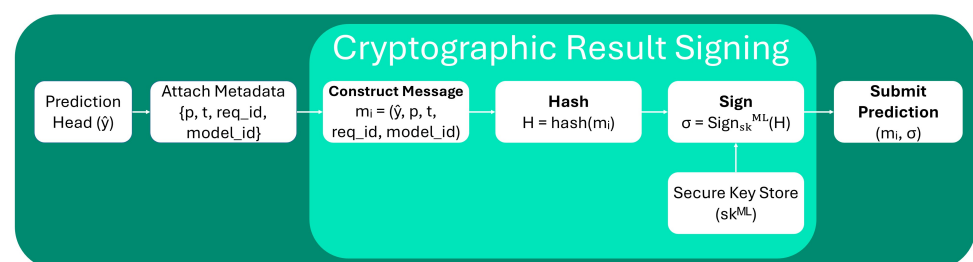


Figure 5. Cryptographic result signing in ODABAC-IoT framework.

If verification succeeds, the decision is accepted and logged without re-executing the full smart contract. If the validation is not successful, the decision is rejected and upgraded

to an on-chain ABAC assessment. Such a handoff of data ensures that adversaries are unable to spoof ML predictions or tamper with them; hence, meritorious trust in off-chain inference outcomes is established.

3.2.4. Periodic Model Verification and Retraining

To avoid concept drift and guarantee long-term dependability, the system incorporates periodic model verification and retraining mechanisms. A small fraction $\alpha\%$ of high-confidence decisions are arbitrarily selected and checked on-chain by re-executing the full ABAC policy evaluation as shown in Equation (23)

$$\text{AuditSet} = \{ m_i \mid \text{Random}(i) < \alpha \} \quad (23)$$

This ensures that even predictions accepted off-chain are occasionally double-checked for correctness. If the ML prediction outcome y^i do not match the ABAC ground-truth decision y_i^{ABAC} , the inconsistency is logged as shown in Equation (24)

$$\Delta_i = \mathbb{K}(\hat{y}_i \neq y_i^{\text{ABAC}}) \quad (24)$$

where

- $\Delta_i = 1$: misclassification detected.
- $\Delta_i = 0$: prediction validated.

If the inconsistency exceeds a threshold β , the ML model is flagged for retraining. Equation (25) is used to check the inconsistency range.

$$\frac{\sum_i \Delta_i}{|\text{AuditSet}|} \geq \beta \quad (25)$$

This ensures adaptation to evolving user/device behaviors and changing grid conditions. The ML model is retrained using both historical CouchDB records and newly audited ABAC data.

Figure 6 shows the process where a fraction of off-chain decisions is recomputed (shown using dotted line) using the ABAC engine. Inconsistency discovery guarantees mismatches are logged, while exceptionally, the inconsistency threshold triggers retraining. The updated dataset is then used to redeploy the ML model.

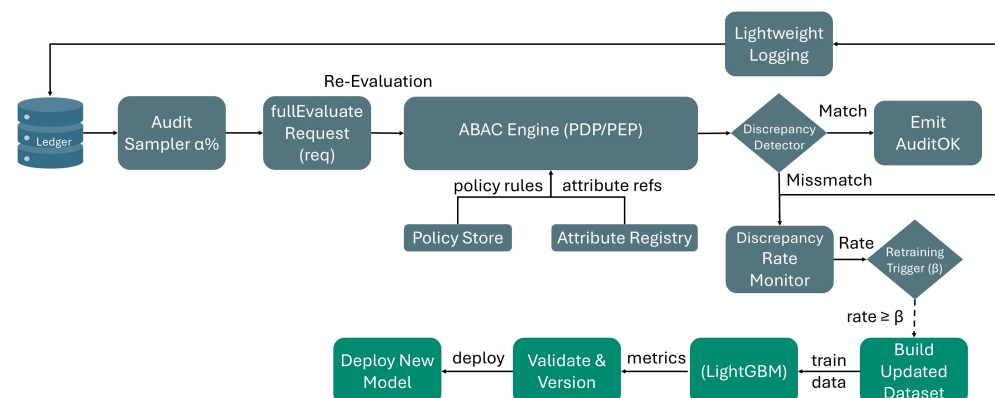


Figure 6. On-chain auditing and model retraining workflow.

3.3. On-Chain Processing

In [2] the on-chain processing was distributed across Hyperledger Fabric and Hyperledger Besu. It utilized four contracts on fabric for user (UAC), device (DAC), resource (RAC) and certificate management contracts (CMC) which were responsible for maintain-

ing attribute registries and cryptographic trust, whereas Besu maintained access policy (APC) and access decision (ADC) contracts. This design ensured immutability, auditability, and zero-trust enforcement, but the continuous on-chain assessment of attributes for every request gave rise to computational overhead and scalability challenges. Before contributing to the framework, a blockchain-based registration process is necessary for every type of entity, as shown in Figures 7 and 8. For instance, devices such as smart meters, HVAC systems, etc., are onboarded through the DAC. Each device is bound to a certificate issued by the CMC, guaranteeing authenticity. Likewise, users are registered through the UAC and validated by CMC-issued certificates. Together, these registries guarantee that only authenticated users and certified devices can issue access requests, while failed authorization requires repeating the process.

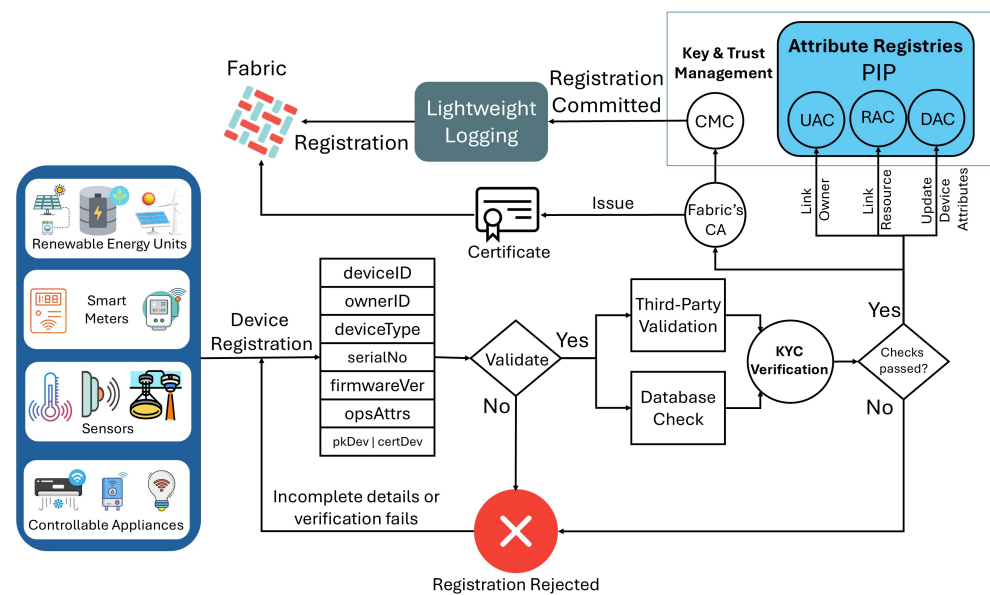


Figure 7. Device registration via the DAC with certificate validation through the CMC.

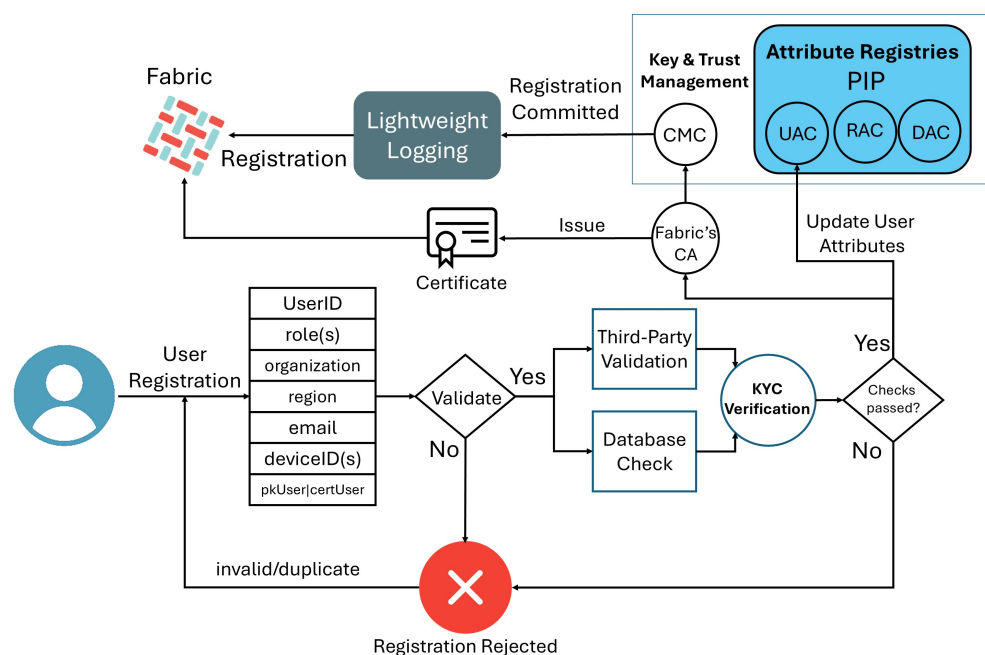


Figure 8. User registration via the UAC with certificate validation through the CMC.

As illustrated in Figure 9, the RAC complements the UAC and DAC by maintaining dynamic, resource-specific attributes such as tariff schedules, demand response signals, and grid thresholds. These three attribute stores collectively serve as the Policy Information Point (PIP). Policies themselves are authored in the APC and deployed on Hyperledger Besu, where they define contextual rules such as time-of-day restrictions, energy-trading limits, or device-specific authorizations.

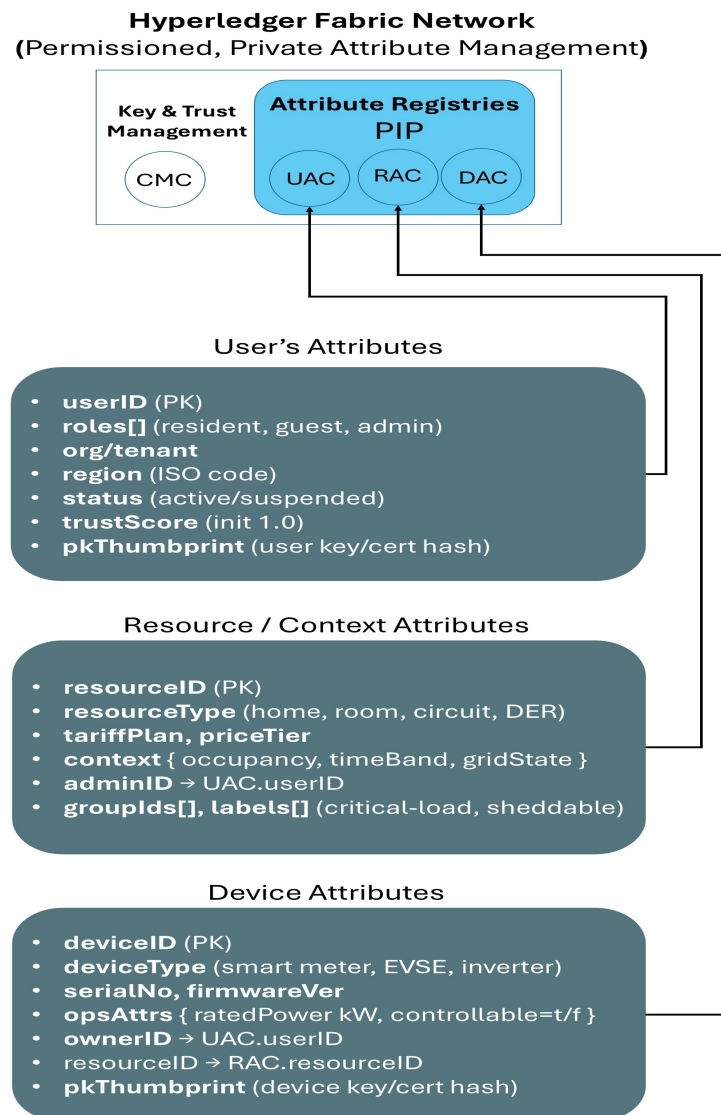


Figure 9. Attribute store schematic (PIP on Fabric).

In ODABAC-IoT, the architecture preserves this contract structure but optimizes it by introducing a LightGBM-assisted off-chain inference stage. The ADC is the primary point of modification, extended with two distinct entry paths:

1. **submitPrediction()** which accepts high-confidence LightGBM predictions signed by the off-chain inference node and verifies their cryptographic integrity before committing a minimal decision record on-chain.
2. **fullEvaluateRequest()** which executes the complete ABAC workflow across the UAC, RAC, DAC, and APC for low-confidence or anomalous requests.

Figure 10 shows the hybrid blockchain framework integration for on-chain processing. This double pathway lessens redundant on-chain assessments while guaranteeing that ambiguous access still experiences full blockchain endorsement.

The selective on-chain verification mechanism in ODABAC-IoT operates as a confidence-driven event-triggered process. Specifically, access requests are only escalated to the blockchain when the prediction confidence falls below a predefined threshold or when periodic auditing conditions are met. This behavior is conceptually similar to event-triggered control mechanisms, where system updates are executed only when certain conditions are violated. By avoiding continuous on-chain evaluation, the framework reduces communication and computational overhead while maintaining security through conditional verification.

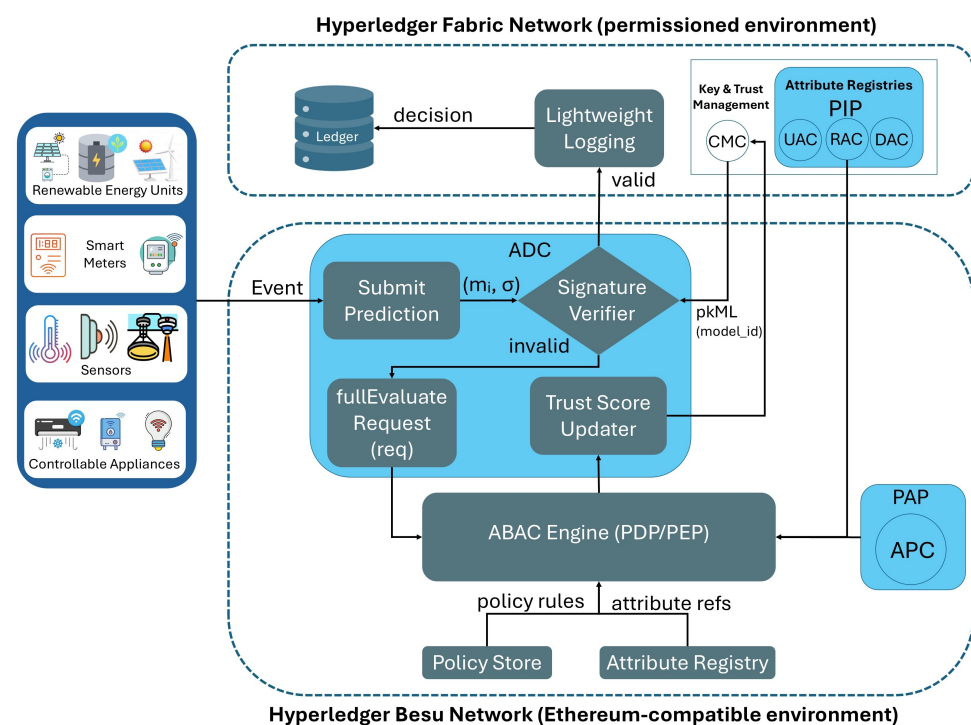


Figure 10. On-chain processing in the proposed ODABAC-IoT framework.

The two new entry points added to the smart contract distinguishing between high-confidence and low-confidence requests are:

submitPrediction()

- Accepts a decision tuple $mi = (y^i, p_i, t_i, \sigma_i)$ from the ML engine.
- Verifies the cryptographic signature σ_i using the ML node's public key.
- Validates confidence metadata ($p_i \geq \tau$).
- Logs the result to the ledger without executing the full ABAC policy.

Formally:

$$\text{if } \text{Verify}_{pk_{ML}}(H(m_i), \sigma_i) = \text{True and } p_i \geq \tau \Rightarrow \text{Ledger.Log}(H(m_i))$$

fullEvaluateRequest()

- Invokes the traditional ABAC evaluation engine.
- Used for:
 - (a) Low-confidence predictions ($p_i < \tau$);
 - (b) Randomly sampled audit requests.

This guarantees that the blockchain continuously has a dependable contingency to have firm on-chain policy checks.

To reduce the quantity of data on the blockchain ledger, the RAC and ADC were also streamlined to store records of decision outcome, cryptographic hashes, and minimal contextual metadata:

- Decision hash $H(m_i)$ instead of the full feature vector.
- Minimal attribute metadata (e.g., device ID, request type).

Formally:

$$\text{LedgerEntry} = (H(m_i), \text{DeviceID}, \text{Action}, \text{Timestamp})$$

This improves the query efficiency while maintaining immutability and transparency with a minimized ledger.

Lastly, the trust recalibration mechanism was optimized within the ADC and CMC. Any off-chain prediction deviating from the verified outcomes of ABAC triggers modification to the device's or inference node's trust score, guaranteeing continuous zero-trust arrangement. Whenever the off-chain prediction y^i conflicts with ABAC decision y_i^{ABAC} , the trust score is updated:

$$\text{Trust}_{\text{new}} = \text{Trust}_{\text{old}} \delta \cdot \Delta_i$$

where $\Delta_i = 1$ if a discrepancy is detected, and δ is the penalty factor.

This guarantees that ML-based optimizations do not weaken security guarantees but are instead incorporated into the dynamic trust cycle.

The on-chain processing layer as shown in Figure 11 complements the off-chain prediction engine by ensuring verifiability, immutability, and trust. Through selective logging, cryptographic validation, and trust score updates, it provides a secure fallback that balances efficiency with accountability. This hybrid design not only reduces blockchain computation costs but also strengthens resilience, making the ODABAC-IoT framework scalable, auditable, and adaptable in dynamic IoT environments.

Scalability of the proposed ODABAC-IoT framework is achieved through its hybrid architecture, which minimizes reliance on computationally expensive on-chain operations. As the number of IoT devices and access requests increases, the majority of high-confidence decisions are processed off-chain using the LightGBM inference engine, thereby preventing congestion in the blockchain layer. Only a subset of uncertain or anomalous requests are escalated to on-chain evaluation, ensuring that blockchain workload grows sub-linearly with system scale. Furthermore, the use of distributed edge nodes for off-chain processing enables parallel evaluation of access requests, improving throughput in large-scale deployments such as smart communities or grid-level systems.

The proposed ODABAC-IoT framework is designed to align with key regulatory requirements related to data governance. Blockchain-based logging ensures immutable and auditable records of all access decisions, supporting transparency and accountability. Data retention is handled through selective on-chain storage, where only essential metadata and cryptographic hashes are recorded, minimizing exposure of sensitive information. User consent and privacy are preserved by restricting access to contextual attributes and enforcing policy-based authorization through ABAC rules. Furthermore, the use of off-chain processing reduces the quantity of personal or sensitive data stored on-chain, which is beneficial for compliance with modern data protection principles. The hybrid on-chain enforcement process for handling high-confidence and low-confidence access requests is formally described in Algorithm 2.

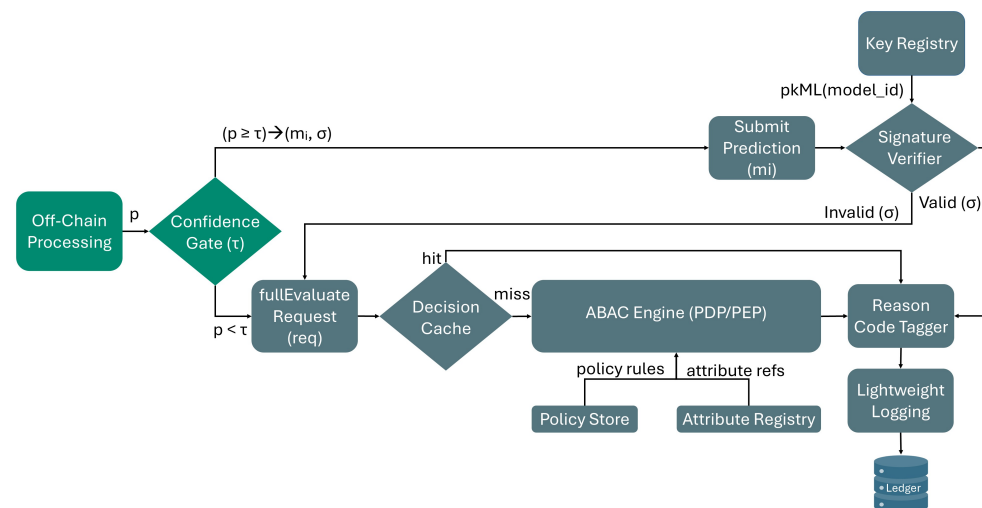


Figure 11. On-chain processing layer of ODABAC-IoT framework.

Algorithm 2 Smart- contract logic for hybrid ABAC enforcement

Input:

- Decision tuple from ML engine:

$$m_i = \langle \hat{y}_i, p_i, t_i, \sigma_i \rangle$$

where $\hat{y}_i$ = ML decision, p_i = confidence, t_i = timestamp, σ_i = signature.

- Confidence threshold τ .
- Audit rate $\alpha \in [0, 1]$.
- ABAC policy engine (on-chain).

Output:

- Final access decision logged on the blockchain ledger.

Procedure

- Function** submitPrediction(m_i)
 - Verify σ_i using ML node's public key.
 - If signature invalid $\Rightarrow$ reject request.
 - If $p_i \geq \tau$:
 - Append $H(m_i)$ to ledger.
 - Return decision = $\hat{y}_i$.
 - Else: Call fullEvaluateRequest(ϕ_i).
 - Function** fullEvaluateRequest(ϕ_i)
 - Execute ABAC policy engine on attribute vector ϕ_i .
 - Obtain ground truth decision u_i .
 - Log (u_i, ϕ_i, t_i) to the ledger.
 - If $u_i \neq \hat{y}_i$:
 - Trigger trust recalibration event.
 - Mark (ϕ_i, u_i) for retraining dataset.
 - Return decision = u_i .
 - Audit Sampling**
 - For each high confidence request ($p_i \geq \tau$):
 - Draw $q \sim U(0, 1)$.
 - If $q < \alpha$, override and call fullEvaluateRequest(ϕ_i) (audit).
-

4. Scenario: Smart-Home Energy Access Request

To demonstrate the functioning of the proposed framework, consider a scenario in which many devices, such as a smart meter, HVAC system, solar inverter, and battery, communicate with the HEMS and other utility networks. The devices often make requests for actions such as fetching information on usage, load control, and energy trading.

In [2] devices were placed into three categories of High Security (smart grid controller), Moderate Security (thermostat), and Low Security (light bulb); these were all identified as an equivalent level of security. ODABAC-IoT extends this classification to realistic smart-home energy devices, as summarized in Table 2. These devices are associated with AC attributes, enriched with temporal and contextual features assembled off-chain for ML processing, and assigned a criticality level consistent with zero-trust enforcement.

Upon creation of a new request (such as a smart meter sending data on usage), the IoT Device Layer gathers context-based attributes, including attributes related to device identification, location, timing of request, usage pattern, and user preference. The feature vector created from these attributes undergoes processing in the Off-chain ML Processing Layer using the LightGBM classifier, which generates a probability for that request being aligned with acceptable behavior.

Table 2. IoT device attributes and contextual features in ODABAC-IoT scenario.

Device	Primary Operations	Access Control Attributes [2]	Temporal/Contextual Variables (ODABAC-IoT)	Device Criticality (Security Class)	Measuring Unit/Normalization
Smart Meter	Report consumption data; request billing info	Device ID, User ID, Role, Location, Certificate validity	Time of request, reporting interval, tariff period (peak/off peak)	High Security—mandatory for billing and grid balancing	Energy (kWh), normalized by daily average load
HVAC System	Adjust temperature; demand scheduling	Device ID, User role, Authorized function (cooling/heating)	Time of day, indoor temperature, occupancy status, external weather	Moderate Security—comfort-oriented, not grid-critical	Power (kW), normalized by rated capacity
Solar Inverter	Inject PV power; request synchronization	Device ID, Location, Policy ID, Certificate validation	Solar irradiance, time of day, seasonal variation, grid demand	High Security—impacts grid injection stability	Power (kW), normalized by installed PV capacity
Energy Storage Unit	Charge/discharge scheduling; backup provisioning	Device ID, State of Charge (SoC), Resource policy	Time of request, SoC threshold, demand response signal, grid status	Critical/High Security—reliability and grid balancing	Energy (kWh), normalized by maximum capacity
HEMS Controller	Orchestrates device actions; interacts with utility	Controller ID, User role, Policy ID, Certificate status	Energy price signals, demand response events, household load forecast	Very High Security—central trust anchor	Aggregated load (kWh), normalized vs. baseline household demand

The experiments were conducted on an Intel Core i7 workstation with 16 GB RAM and 256 GB SSD running Ubuntu 22.04 LTS (Dell Inc., Round Rock, TX, USA).

- **High-Confidence Case:** If the probability surpasses a predefined threshold (e.g., $\geq 95\%$), the decision is cryptographically signed and submitted to the ADC through the *submitPrediction()* function. The ADC confirms the signature, records the decision hash, and grants access without invoking full on-chain evaluation, as shown in Figure 12.
- **Low-Confidence Case:** If the confidence score is below the threshold, or if anomalies are noticed, the request is diverted to the *fullEvaluateRequest()* function. A complete

ABAC process runs on-chain via the UAC, DAC, RAC, and APC, ensuring rigorous verification and preserving zero trust enforcement, as shown in Figure 13.

Example: A smart meter performing a standard request at its regular reporting interval is quickly confirmed through off-chain LightGBM inference with negligible blockchain costs. On the contrary, if the same smart meter tries to access the system at an odd hour or reports strange consumption, the low-confidence classification sets off complete on-chain ABAC validation.

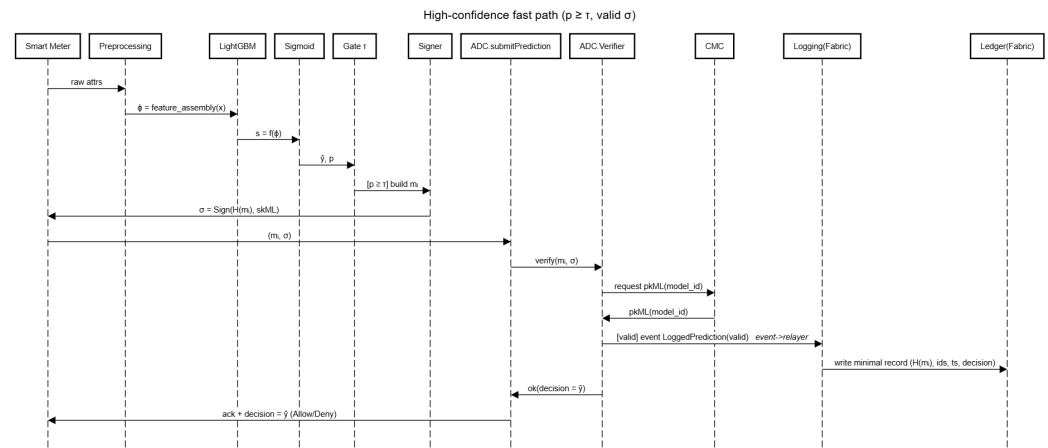


Figure 12. High-confidence fast path.

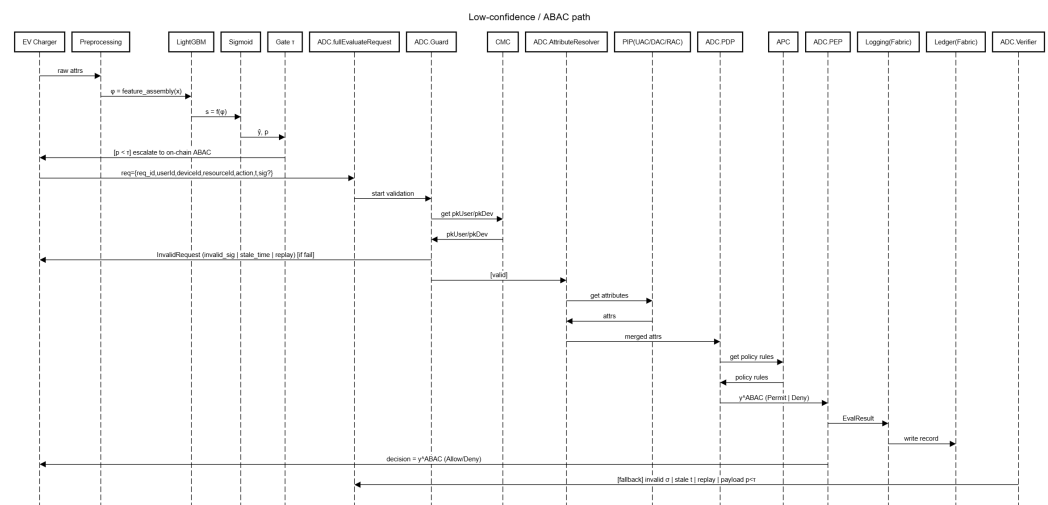


Figure 13. Low-confidence ABAC path.

Workflow Summary

- Request Generation:**
IoT device sends an access request with contextual attributes.
- Off-Chain Preprocessing:**
Attributes are normalized, validated, and passed to the ML prediction engine.
- Prediction and Thresholding:**
 - High confidence** → *submitPrediction()* on-chain.
 - Low confidence** → *fullEvaluateRequest()* on-chain.
- On-chain Enforcement:**
Blockchain either logs the prediction result (high confidence) or executes the full ABAC evaluation (low confidence).

5. **Periodic Verification:** Random samples of high-confidence cases are verified on-chain to ensure ML model reliability.
6. **Retraining:** ML model retrains periodically with updated labeled data from blockchain verified requests.

5. Results and Discussion

5.1. Experimental Setup

The implementation and evaluation of the proposed ODABAC-IoT framework were simulated and evaluated on a Ubuntu 22.04 LTS operating system with 16 GB of RAM, an Intel Core i7 processor, and 256 GB of SSD storage, ensuring efficient execution and scalability.

The dataset used in this study was generated based on simulated IoT access scenarios and augmented using realistic attribute distributions derived from typical smart-home environments.

The hybrid blockchain infrastructure was emulated on a private network with the help of Hyperledger Fabric (v2.5) to manage the registries of users, devices, and resources, and Hyperledger Besu (v23.4.1) to enforce access control rules and run smart contracts. The testbed for smart-home energy management comprised a typical collection of IoT devices like smart meters, HVAC controllers, solar inverters, and energy storage units generating synthetic access requests under dynamically varied contextual conditions (e.g., peak/off peak hours, occupancy status, renewable energy availability).

In order to facilitate predictive processing off-chain, the ML inference engine based on LightGBM was developed in Python (v3.10) and utilized Docker (v20.10) for deployment on edge nodes within the simulated environment. The engine was linked with Apache CouchDB (v3.2) for keeping historical requests and managing training datasets generated through simulation. This setup offered a realistic environment for testing system latency, transaction volume, energy consumption, and storage overhead. A detailed description can be found in Table 3, which includes the hardware specifications, software tools, and IoT device testbeds that are necessary for assessing the performance and scalability of the proposed system.

Table 3. Components description.

Component	Description & Purpose
Ubuntu 22.04 LTS + Hardware Specs	Provides stable computing environment; sufficient memory/processing capacity ensures realistic performance metrics.
Hyperledger Fabric	Manages dynamic attribute storage and updates, critical for context-aware ABAC policy enforcement.
Hyperledger Besu	Hosts smart contracts and enforces ABAC logic; supports private chain execution contexts for smart homes.
IoT Device Testbed	Simulates real-world energy IoT devices; generates diverse contextual access requests covering energy-relevant scenarios.
Python + LightGBM (Docker)	Mimics edge-level ML deployment; containerization highlights system deployability and reproducibility.
CouchDB Storage	Enables offline training data retention and history logging, essential for ML model performance and auditability.

5.2. Transaction Volume Reduction

One of the main aims of the ODABAC-IoT framework is to cut down the transactions run on the blockchain to a minimum, thus lessening the computational overhead, storage growth, and the energy costs linked with them. In the typical on-chain ABAC system, the full policy evaluation via smart contracts is mandatory for every access request, thereby leading to one transaction per request. Thus, for 10,000 requests that were simulated, the on-chain system marked almost 10,000 transactions, which mirrors the linear growth problem that has been a part of the traditional blockchain-based ABAC models. In contrast, the former suggested hybrid blockchain framework [2] gave rise to partial optimizations by distinguishing between static and dynamic attributes thus limiting the total transactions to nearly 7000, a 30% reduction. However, this degree of optimization turns out to be inadequate in the long run when the numbers of IoT devices and requests for access keep growing. By contrast, ODABAC-IoT introduces an off-chain LightGBM pre-evaluation layer that filters out high-confidence decisions, submitting only essential verification records to the blockchain. Table 4 summarizes the transaction volume across different request loads for the baseline ABAC model, hybrid Fabric–Besu model, and the proposed ODABAC-IoT framework. As a result, for the same 10,000 requests, the number of on-chain transactions was reduced to nearly 3500, yielding a 65% reduction compared to on-chain ABAC and 50% compared to hybrid blockchain approaches. This significant reduction, as shown in Figure 14, demonstrates the effectiveness of integrating lightweight ML inference in alleviating the transaction burden on the blockchain, directly contributing to enhanced scalability, reduced ledger growth, and improved responsiveness in smart HEMSs.

The baseline ABAC model shows near linear growth as every request triggers full on-chain evaluation. The hybrid Fabric–Besu model reduces transactions moderately by distributing processing across chains. The proposed ODABAC-IoT achieves the most significant reduction by resolving high-confidence requests off-chain, cutting transaction volume by over 50% under high request loads.

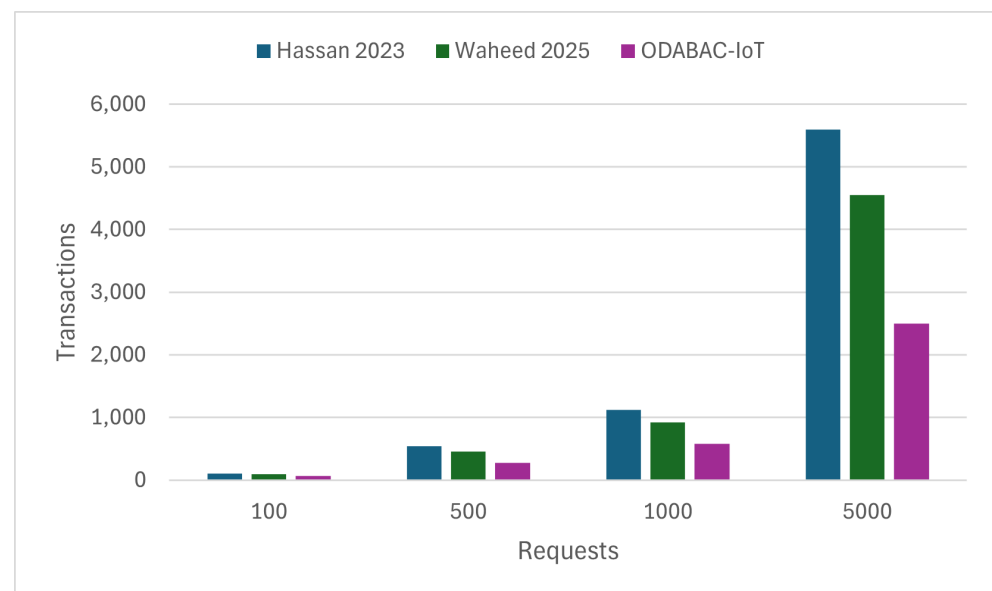


Figure 14. Transaction volume reduction across baseline ABAC ([1]), hybrid Fabric–Besu ([2]), and ODABAC-IoT.

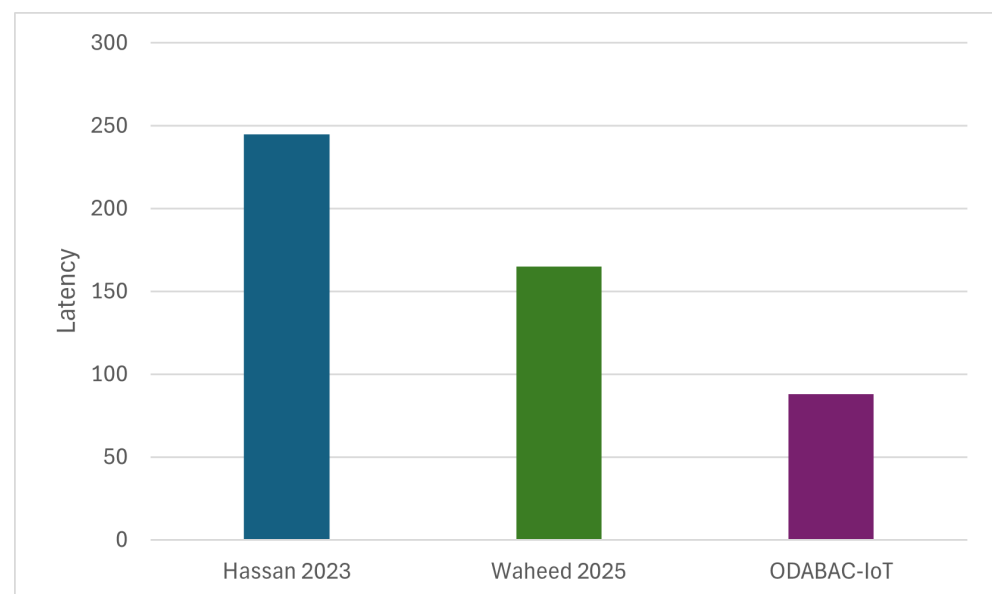
Table 4. Transactions across request loads for baseline ABAC [1], hybrid Fabric-Besu [2] and ODABAC-IoT (Proposed).

Number of Requests	Baseline ABAC [1]	Hybrid Fabric Besu [2]	ODABAC-IoT (Proposed)
100	105	95	70
500	540	460	280
1000	1120	920	580
5000	5600	4550	2500

5.3. Latency Improvement

In IoT-based energy management systems, latency is essential to performance; access requests must typically be verified almost instantly in order to facilitate dynamic operations, such as demand response (load shifting), renewable energy scheduling, etc. The baseline on-chain ABAC system takes each request and performs a full policy evaluation across all smart contracts plus consensus validation, which results in an average decision latency of approximately 245 milliseconds per request. Because of this relatively high latency, decision latency is a critical issue in scenarios requiring a timely response, such as turning on a storage unit in response to fluctuations in grid frequency or controlling HVAC loads based upon real-time tariff changes.

The previously proposed hybrid blockchain framework in [2] helped to alleviate this burden to a certain extent by optimizing attribute management. This resulted in a reduction in average latency to 165 ms per request, which translates to a 32.6% improvement over the pure on-chain model. However, the responsiveness was still not sufficient for latency-constrained smart-grid applications. The proposed ODABAC-IoT framework improves responsiveness significantly, as depicted in Figure 15. This is because high-confidence access requests are offloaded to the off-chain LightGBM inference engine.

**Figure 15.** Latency reduction across baseline ABAC ([1]), hybrid Fabric-Besu ([2]), and ODABAC-IoT.

This selective approach reduced average latency to 88 ms per request, achieving a 64.1% reduction compared to the baseline on-chain ABAC and a 46.7% improvement over the earlier hybrid framework. The results highlight the importance of integrating lightweight off-chain ML processing, which enables ODABAC-IoT to meet the stringent real-time requirements of smart energy systems while maintaining zero-trust security.

guarantees. Table 5 shows the latency reduction and comparison across the baseline ABAC model, hybrid Fabric–Besu model, and the proposed ODABAC-IoT framework under varying request loads.

The average latency values represent overall system performance under comparable experimental conditions and are used to summarize the latency improvements across different frameworks.

Table 5. Average latency per request and percentage reduction relative to baseline ABAC.

Method	Latency (ms/request)	Reduction vs. Baseline
Hassan 2023 [1]	245	N/A
Waheed 2025 [2]	165	32.6%
ODABAC-IoT (Proposed)	88	64.1%

Figure 16 describes latency improvement with rising request loads for baseline ABAC, hybrid Fabric–Besu, and ODABAC-IoT. In the baseline system, there is a sharp increase in latency with each rising request, since each request is subjected to full on-chain evaluation. In the hybrid system, the rate of increase in latency is reduced with the help of distributed processing. In ODABAC-IoT, the maximum improvement in latency is achieved with the help of off-chain filtering of high-confidence requests, with an improvement of over 70% in latency with rising request levels. Table 6 describes the values of latency for the baseline ABAC, hybrid Fabric–Besu, and ODABAC-IoT systems with varying levels of requests.

Latency values represent the average time required to process access requests under identical workload conditions, measured as the time elapsed between request generation and final authorization decision.

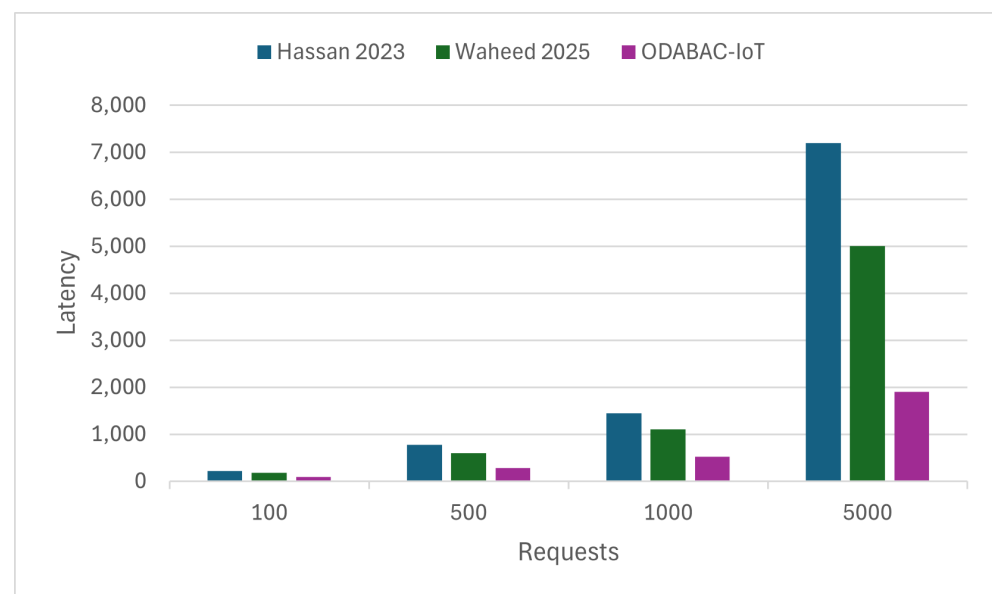


Figure 16. Latency variation with increasing request loads for Baseline ABAC ([1]), Hybrid Fabric–Besu ([2]), and ODABAC-IoT.

Table 6. Latency across baseline ABAC ([1]), hybrid Fabric-Besu ([2]), and ODABAC-IoT.

Number of Requests	Baseline ABAC [1] (ms)	Hybrid Fabric-Besu [2] (ms)	ODABAC-IoT (Proposed) (ms)
100	220	280	95
500	780	600	280
1000	1450	1100	520
5000	7200	5000	1900

5.4. Energy-Efficiency Outcomes

The design of ODABAC-IoT is motivated not just by the desire to improve computational efficiency but also by the need for energy sustainability; both of these are of utmost importance when deploying technologies in smart homes and smart grids. One of the main drawbacks of implementing blockchain-based systems as AC mechanisms is the additional computational and storage overhead they create for all of the nodes that must continue to provide maintenance to the network, resulting in additional energy consumption. Following the methodology outlined by Waheed et al. [2], the average computational energy cost of a blockchain transaction in a lightweight, permissioned environment can be conservatively estimated at $E_tx = 0.01$ kWh, considering both consensus operations and smart contract execution. Figure 17 is a rough approximation based on previous work and was considered in order to conduct comparisons since there can be differences in actual consumption due to the underlying hardware and consensus algorithms employed in various networks.

In order to perform a sensitivity analysis and examine the validity of the energy consumption model, the energy per transaction (E_tx) factor was varied within $\pm 20\%$ from its default value. The result showed that while the absolute value of energy consumed changed proportionally with E_tx , the relative efficiency gains of ODABAC-IoT were still maintained. Hence, it can be concluded that the observed benefits are not dependent on the value of E_tx .

In the baseline on-chain ABAC model, every access request is fully evaluated on-chain, leading to the highest energy demand. For 100,000 simulated requests, this translated to approximately $Y_1 = 1000$ kWh. The previously proposed hybrid blockchain framework [2], which streamlined attribute handling, reduced the number of on-chain executions by about 30%, lowering total energy usage to around $Y_2 = 700$ kWh. By contrast, the proposed ODABAC-IoT framework achieved a far greater reduction by selectively delegating high-confidence access decisions to the off-chain LightGBM inference engine, while cryptographically verifying the results on-chain. This selective strategy reduced the number of blockchain transactions by nearly 65%, resulting in an energy consumption of only $Y_3 = 350$ kWh per 100,000 requests, a 65% reduction compared to on-chain ABAC and 50% compared to hybrid blockchain approaches. Assuming linear scaling, the energy consumption per request would be approximately 0.01 kWh/request, and total energy would scale proportionally with the number of access requests.

When scaled to a realistic household deployment with ~ 20 devices generating approximately 1000 access requests per day, ODABAC-IoT reduces daily blockchain energy use from 10 kWh/day in the on-chain baseline to 3.5 kWh/day, yielding a daily saving of $Z \approx 6.5$ kWh.

These savings represent the equivalent of continuously running an efficient 65 W LED lighting system for 100 h. This represents small savings for an individual home; however, when scaled to a large number of households (e.g., 1000 homes in a smart-grid-based community setting), the savings can represent significant contributions to sustainability.

Assuming an average grid emission factor, the reduction in energy consumption also implies a proportional decrease in carbon dioxide emissions; however, a detailed emission model is beyond the scope of this study. Table 7 provides a comparison of the energy usage of the baseline on-chain ABAC protocol, hybrid blockchain protocol, and the proposed ODABAC-IoT protocol per 100,000 requests. This table highlights the significant savings achieved by the ODABAC-IoT protocol.

Table 7. Comparative energy consumption of access control frameworks (per 100,000 requests).

Framework	Energy Consumption (kWh)
On-chain ABAC [1]	1000
Hybrid Blockchain [2]	700
ODABAC-IoT (Proposed)	350

Figure 17 demonstrates how these frameworks use energy to process 100,000 requests, with an assumption of linear relationship between the number of transactions processed and the energy required to process transactions, with a fixed energy model per transaction ($E_t x$) being used when calculating the energy consumption per transaction. The on-chain ABAC model experiences the highest level of growth in demand for energy, while the hybrid blockchain model experiences partial reductions versus the ABAC model. The ODABAC-IoT model continues to have a low level of usage and has a much flatter slope versus the ABAC model. The results demonstrate that ODABAC-IoT enhances operational efficiencies and supports the environmental sustainability of energy-efficient IoT systems in their contributions toward achieving smart-energy-system sustainability goals.

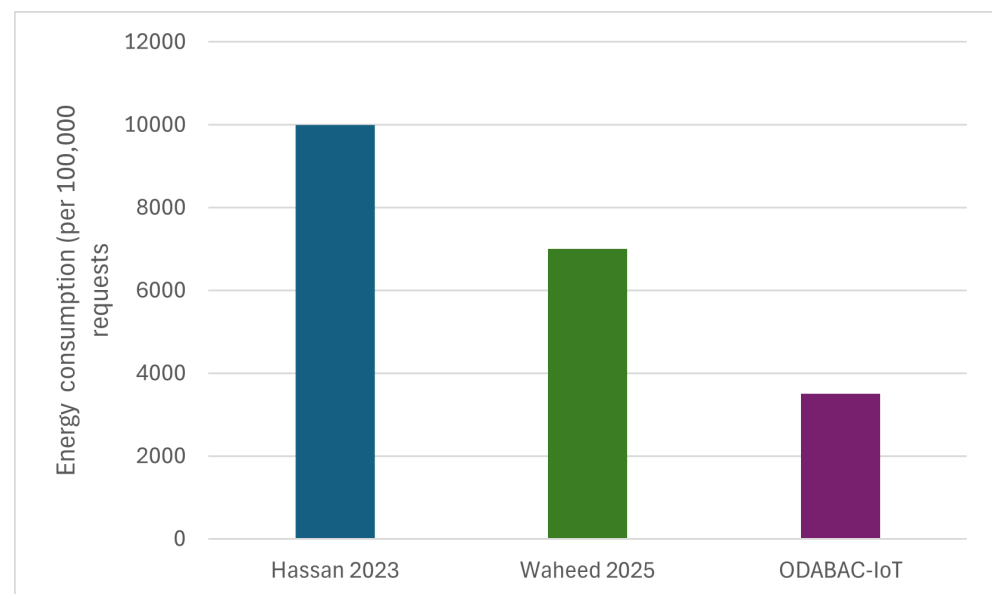


Figure 17. Energy consumption comparison for Baseline ABAC ([1]), Hybrid Fabric-Besu ([2]), and ODABAC-IoT.

The ODABAC-IoT framework enhances the performance of transactions by performing off-chain inference using LightGBM. This method employs an efficient leaf-wise gradient-boost algorithm and histogram-based decision trees which allow for faster training and lower memory requirements when compared to traditional ensemble algorithms.

These algorithmic benefits are reflected in the reduced number of CPU cycles per access request evaluation, particularly when implemented on small edge nodes. Consequently, edge devices experience a power consumption decrease of approximately 20% (based on

comparative computational profiling) during inference relative to baseline off-chain models. This two-layer minimization, both at the blockchain level and at the edge-processing level, shows that ODABAC-IoT can achieve end-to-end AC pipeline energy savings, which justifies its use as a sustainable and green computing architecture in smart-home energy systems. Figure 18 indicates that LightGBM inference requires less computational resources than other ensemble models, reducing edge-node power consumption by as much as 22% and supplementing the energy savings of the blockchain side. The results shown here were obtained through analytical modeling and simulation-based evaluation. The comparison of the different algorithms in this study was based on the same set of workloads and compared based on the computational efficiency of each algorithm, considering its model complexity and inference cost. Figure 18 shows the results of this empirical evaluation.

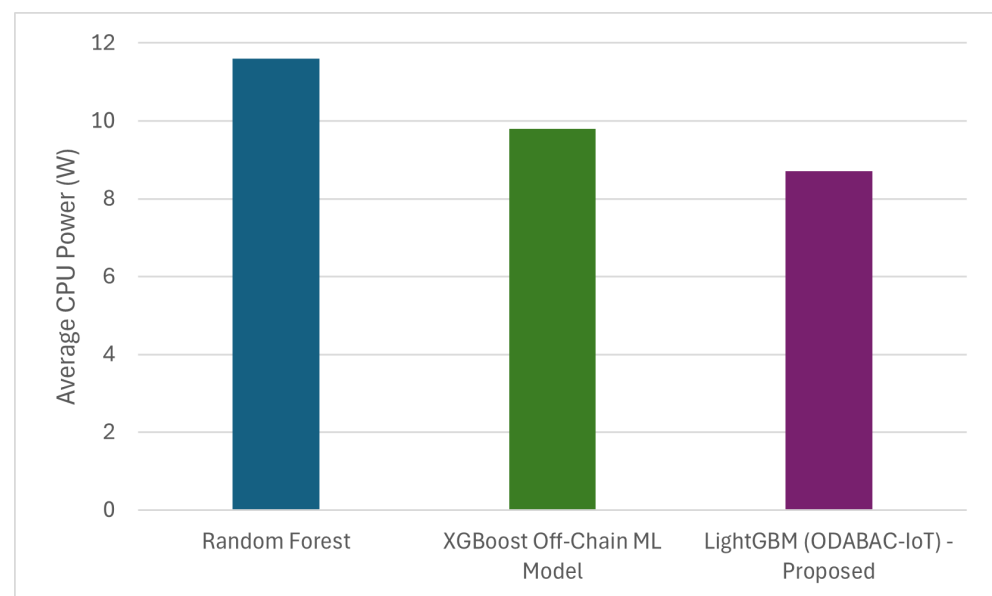


Figure 18. Comparative edge-node power consumption (per 1,000 inference requests).

5.5. Security, Trust, and Standards Compliance

Beyond performance and efficiency gains, the ODABAC-IoT framework is explicitly designed to preserve zero-trust principles and ensure compliance with established and emerging standards in the energy domain. A core element of this design is the cryptographic signing of all high-confidence off-chain ML predictions, which guarantees their authenticity before submission to the blockchain. The system requires random auditing of its predictions because this process protects the off-chain ML inference engine from unauthorized operation. The combination of cryptographic integrity together with randomized auditing creates a reliable trust system that protects against adversarial attacks while allowing operational efficiency. ODABAC-IoT meets multiple international standards through its compliance with various regulatory frameworks.

- **NISTIR 7628 [53]**
ODABAC-IoT fulfills the NIST requirement for continuous monitoring, integrity validation, and accountability by ensuring that each AC decision is verified through immutable logs.
- **IEC 62351 [54]**
ODABAC-IoT implements the communication and system security standard for power systems, IEC 62351, using cryptographic guarantees of integrity, authenticity and confidentiality to protect data exchange within energy management systems as part of

an overall focus on securing IoT-enabled devices through ODABAC-IoT's decision-making process.

- **IEEE 2030.5 [55]**

The context-aware decision-making and secure logging elements of IEEE 2030.5 (SEP2.0) inherently support the interoperability/security requirements for distributed energy resources as specified within this smart energy profile standard.

Moreover, the auditable and tamper-proof decision logs that can be maintained through ODABAC-IoT can help in establishing a robust compliance strategy with FERC/NERC Critical Infrastructure Protection (CIP) requirements, which emphasize the need for transparent, traceable, and verifiable AC mechanisms for critical infrastructures in the energy domain. Thus, by establishing itself as a compliance-ready framework, ODABAC-IoT is capable of supporting a wide range of secure smart homes, microgrid systems, and grid-integrated IoT environments.

Table 8 maps the relevant international standards to the specific contributions of ODABAC-IoT, highlighting how each security, trust, and compliance requirement is addressed by the framework.

Table 8. Standards mapping and ODABAC-IoT contributions.

Standard/Regulation	Key Requirement	Contribution of ODABAC-IoT
NISTIR 7628 [53]	Continuous monitoring, integrity validation, and auditability of smart-grid systems	Cryptographic signing of ML decisions and periodic on-chain auditing ensure continuous verification and tamper proof logging of AC events.
IEC 62351 [54]	Confidentiality, integrity, and authentication of communication in energy management systems	All access requests and ML predictions are cryptographically verified before execution, preserving data integrity and authentication.
IEEE 2030.5 [55]	Secure, interoperable communication and control of distributed energy resources (DERs)	Context-aware, zero-trust ABAC policies with secure decision logs enable interoperability and trustworthy DER integration.
FERC/NERC CIP [56]	Transparent, traceable, and auditable AC mechanisms for critical energy infrastructures	Immutable, auditable blockchain logs of AC decisions support regulatory compliance for grid-level security.

5.6. Sustainability Impact

The ODABAC-IoT framework aligns with UN SDG 7 (Affordable and Clean Energy) by lowering blockchain-related electricity demand by $\approx 65\%$. At the community scale (e.g., 1000 homes), this equates to ≈ 6.5 MWh in annual savings. The reduced on-chain workload also extends node hardware life, indirectly cutting embodied carbon. Hence, ODABAC-IoT contributes to energy-aware and carbon-efficient smart-grid ecosystems.

6. Discussion

The findings from this study demonstrate that the proposed ODABAC-IoT framework achieves a well-calibrated balance between efficiency, accountability, and compliance readiness. From a performance point of view, the use of off-chain LightGBM inference has a significant impact, reducing decision latency and transaction volume on the blockchain, which has a direct impact on reducing the overhead of the system. At the same time, the auditability of the system has not been compromised, as all high-confidence off-chain predictions are cryptographically signed and periodically verified on-chain, ensuring the

integrity of the system is not compromised. From an energy system point of view, the contribution of the proposed system, ODABAC-IoT, towards the green computing movement is direct. The framework achieves energy savings for both homes and communities through its computational optimization, which results in a 65% reduction in expensive on-chain executions. The smart-grid system shows minimal energy savings for individual homes, but the total energy savings become essential when all homes join the smart-grid system. ODABAC-IoT functions as an effective technical solution that improves energy systems through its focus on sustainability. The framework complies with international standards like NISTIR 7628, IEC 62351 and IEEE 2030.5 while it helps organizations meet FERC/NERC CIP compliance obligations. Thus, by incorporating auditable, tamper-proof, and standards-aligned decision logging into its design, ODABAC-IoT is able to assert itself as a compliance-ready framework, one that is appropriate for use in energy-sensitive and critical infrastructures domains, where compliance is a de facto requirement for adoption. In summary, the hybrid design of ODABAC-IoT moves well beyond technical optimization concerns to deliver a holistic solution that improves scalability, energy efficiency, and compliance, making it a progressive architecture for smart homes and grid-integrated energy systems that is secure and sustainable in nature.

7. Conclusions

This paper proposed ODABAC-IoT, a novel hybrid framework using dynamic ABAC to provide secure AC in IoT environments with a focus on being able to handle large numbers of users and devices without sacrificing performance or scalability by using ML to make efficient use of off-chain resources. High-confidence decisions are made by the Light-GBM model based on learned decision patterns, using results it produces, therefore keeping the integrity of the data, while using the blockchain for low-confidence/ambiguous results by doing full on-chain ABAC evaluations. As a result, there are no blockchain transaction overheads when completing a high-confidence ABAC evaluation; however, there is absolutely no loss of verifiability or security. Furthermore, the models are verified and retrained periodically, which allows for long-term sustainability and adaptability as the environment changes. Experimental results demonstrated that the proposed framework achieved up to 65% reduction in blockchain transaction volume, approximately 64% improvement in latency, and 65% reduction in energy consumption compared to conventional on-chain ABAC mechanisms, while maintaining reliable and auditable access control decisions. Moving forward, ODABAC-IoT presents a scalable, cost-effective, and intelligent solution for secure AC in smart-home energy systems, with potential applicability across other critical IoT domains. Future work will focus on improving robustness against adversarial inputs and validating the framework in real-world deployments.

Author Contributions: Conceptualization, U.W. and H.J.; methodology, U.W.; software, Y.M. and N.U.R.M.; validation, M.I.M., A.M.N., M.A. and T.A.J.; formal analysis, U.W. and H.J.; investigation, U.W.; resources, M.I.M.; data curation, U.W., N.U.R.M. and M.A.; writing—original draft preparation, U.W.; writing—review and editing, Y.M., H.J., N.U.R.M. and T.A.J.; visualization, U.W.; supervision, M.I.M. and M.A.; project administration, H.J., A.M.N., M.I.M. and M.A.; funding acquisition, M.I.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to privacy or ethical restrictions.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IoT	Internet of Things
ML	Machine Learning
ABAC	Attribute-Based Access Control
ODABAC-IoT	Optimized Dynamic-Attribute-Based Access Control
RBAC	Role-Based Access Control
LightGBM	Light Gradient-Boosting Machine
AC	Access Control
ABE	Attribute-Based Encryption
AI	Artificial Intelligence
IPFS	InterPlanetary File System
HEMS	Home Energy Management System
HVAC	Heating, Ventilation, and Air-Conditioning
EV	Electric Vehicle
MQTT	Message Queuing Telemetry Transport
HTTPS	Hypertext Transfer Protocol Secure
UAC	User Attribute Contract
DAC	Device Attribute Contract
RAC	Resource Attribute Contract
CMC	Certificate Management Contract
APC	Access Policy Contract
ADC	Access Decision Contract
PIP	Policy Information Point
U_I	User Attribute
V_I	Device Attribute
R_I	Request Attribute
C_I	Contextual Attribute
X_I	Unified Vector

References

- Hasan, M.R.; Alazab, A.; Joy, S.B.; Uddin, M.N.; Uddin, M.A.; Khraisat, A.; Gondal, I.; Urmi, W.F.; Talukder, M.A. Smart Contract-Based Access Control Framework for Internet of Things Devices. *Computers* **2023**, *12*, 240. [\[CrossRef\]](#)
- Waheed, U.; Khan, S.A.; Masud, M.; Jamshed, H.; Jumani, T.A.; Malik, N.U.R. Blockchain Based, Dynamic Attribute Based Access Control for Smart Home Energy Systems. *Energies* **2025**, *18*, 1973. [\[CrossRef\]](#)
- Awan, S.M.; Azad, M.A.; Arshad, J.; Waheed, U.; Sharif, T. A Blockchain Inspired Attribute Based Zero Trust Access Control Model for IoT. *Information* **2023**, *14*, 129. [\[CrossRef\]](#)
- Gono, A.; PISAŠOVIC, I.; Zejda, M.; Landa, J.; Procházka, D. Improving IoT Management with Blockchain: Smart Home Access Control. *Eur. J. Bus. Sci. Technol.* **2024**, *10*, 225–241. [\[CrossRef\]](#)
- Punia, A.; Gulia, P.; Gill, N.S.; Ibeke, E.; Iwendi, C.; Shukla, P.K. A systematic review on blockchain based access control systems in cloud environment. *J. Cloud Comput.* **2024**, *13*, 146. [\[CrossRef\]](#)
- Wang, X.; Yue, X.; Tariq, N.; Sajid, A. Hybrid AI and Blockchain Powered Secure Internet Hospital Communication and Anomaly Detection in Smart Cities. *Processes* **2025**, *13*, 1466. [\[CrossRef\]](#)
- Saleem, J.; Raza, U.; Hammoudeh, M.; Holderbaum, W. Machine Learning Enhanced Attribute Based Authentication for Secure IoT Access Control. *Sensors* **2025**, *25*, 2779. [\[CrossRef\]](#)
- Anjum, M.; Kraiem, N.; Min, H.; Dutta, A.K.; Daradkeh, Y.I.; Shahab, S. Opportunistic access control scheme for enhancing IoT enabled healthcare security using blockchain and machine learning. *Sci. Rep.* **2025**, *15*, 7589. [\[CrossRef\]](#) [\[PubMed\]](#)
- Ben Chehida Douss, A.; Abassi, R. Towards a formal framework for blockchain-based access control in e-health systems. *Ann. Telecommun.* **2026**, 1–25. [\[CrossRef\]](#)

10. Outchakoucht, A.; Abou El Kalam, A.; Es-Samaali, H.; Benhadou, S. Machine Learning Based Access Control Framework for the Internet of Things. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 331–340. [\[CrossRef\]](#)
11. Trabelsi, R.; Fersi, G.; Jmaiel, M. Access control in Internet of Things: A survey. *Comput. Secur.* **2023**, *135*, 103472. [\[CrossRef\]](#)
12. Singh, J.; Rani, S.; Kumar, V. Role based access control (RBAC) enabled secure and efficient data processing framework for IoT networks. *Int. J. Commun. Netw. Inf. Secur.* **2024**, *16*, 91–103. [\[CrossRef\]](#)
13. Díaz, J.P.; Mendoza, F.A. Authorization models for IoT environments: A survey. *Internet Things* **2025**, *29*, 101430. [\[CrossRef\]](#)
14. Kalaria, R.; Kayes, A.S.; Rahayu, W.; Pardede, E.; Salehi Shahraki, A. Adaptive context aware access control for IoT environments leveraging fog computing. *Int. J. Inf. Secur.* **2024**, *23*, 3089–3107. [\[CrossRef\]](#)
15. Ahsan, M.S.; Pathan, A.K. A Comprehensive Survey on the Requirements, Applications, and Future Challenges for Access Control Models in IoT: The State of the Art. *IoT* **2025**, *6*, 9. [\[CrossRef\]](#)
16. Nie, S.; Ren, J.; Wu, R.; Han, P.; Han, Z.; Wan, W.K. Zero Trust Access Control Mechanism Based on Blockchain and Inner Product Encryption in the Internet of Things in a 6G Environment. *Sensors* **2025**, *25*, 550. [\[CrossRef\]](#)
17. Kalarani, S.; Shobana, M.; Uma Shankari, E.; Praveena, B.J.; Shanthi, S.; Ramadevi, R.; Sandiri, R. A secure and cloud based patient management system using attribute based encryption algorithm. *Int. J. Power Electron. Drive Syst.* **2025**, *15*, 2445–2454. [\[CrossRef\]](#)
18. Riad, K. Robust Access Control for Secure IoT Outsourcing with Leakage Resilience. *Sensors* **2025**, *25*, 625. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Ramamurthy, K.; Kulanthaivelu, S.; Jenish, S.A. Future Directions and Emerging Trends of Adaptive Artificial Intelligence in Wireless Sensor Networks, Internet of Things, and Industrial Internet of Things. In *Adaptive AI in Sensor Informatics: Methods, Applications, and Implications*; Ramamurthy, K., Kulanthaivelu, S., Sountharajan, S., Goyal, S.B., Kadry, S., Eds.; Elsevier: Amsterdam, The Netherlands, 2026; pp. 291–308.
20. Mariappan, K.; Ganesan, P.; Jagatheesaperumal, S.K. IoT access control and authentication schemes. In *Internet of Things Security*; Elsevier: Amsterdam, The Netherlands, 2026; pp. 87–106. [\[CrossRef\]](#)
21. Yaqub, N.; Zhang, J.; Khalid, M.I.; Wang, W.; Helfert, M.; Ahmed, M.; Kim, J. Blockchain enabled policy based access control mechanism to restrict unauthorized access to electronic health records. *PeerJ Comput. Sci.* **2025**, *11*, e2647. [\[CrossRef\]](#)
22. Wang, J.; Liu, C.; Zhu, G.; Liu, X.; Xiao, B. Fine grained Trusted Control Methods for IoT Boundary Access. *Scalable Comput. Pract. Exp.* **2025**, *26*, 180–190. [\[CrossRef\]](#)
23. Ramírez-Gordillo, T.; Maciá-Lillo, A.; Pujol, F.A.; García-D’Urso, N.; Azorín-López, J.; Mora, H. Decentralized Identity Management for Internet of Things (IoT) Devices Using IOTA Blockchain Technology. *Future Internet* **2025**, *17*, 49. [\[CrossRef\]](#)
24. Naik, M.; Singh, A.P.; Pradhan, N.R.; Mohanty, S.N.; Rafiq, M.; Ayouni, S.; Khan, M.I. Sustainable NFTs: Enabling green energy adoption through IoT-integrated Blockchain tokenization of carbon offsets. *Int. Commun. Heat Mass Transf.* **2026**, *175*, 110939. [\[CrossRef\]](#)
25. Bhargavi, V.S.; Sawai, N.M.; Chakravarthi, M.K.; Shaik Ibrahim, S.H.; Pérez, L.A.; Tiwari, M. Scalability and Performance of Blockchain-Based Solutions for IoT Applications. In *Interdisciplinary Approaches to AI, Internet of Everything, and Machine Learning*; IGI Global Scientific Publishing: Hershey, PA, USA, 2024; pp. 203–216. [\[CrossRef\]](#)
26. Rouached, M.; Akremi, A.; Macherki, M.; Kraïem, N. Policy Based Smart Contracts Management for IoT Privacy Preservation. *Future Internet* **2024**, *16*, 452. [\[CrossRef\]](#)
27. Cloudin, S.; Robinson Joel, M.; Manikandan, G.; Milcah Blessy, I. Integrating Blockchain, IoT, and 6G Technologies for Secure, Efficient, and Sustainable Smart City Applications: Enhancing Urban Living Through Innovation. In *Building Tomorrow’s Smart Cities With 6G Infrastructure Technology*; IGI Global Scientific Publishing: Hershey, PA, USA, 2024; pp. 483–508. [\[CrossRef\]](#)
28. Ghosh, S.; Mitra, R.; Roy, I.; Gupta, B. Securing RC Based P2P Networks: A Blockchain based Access Control Framework utilizing Ethereum Smart Contracts for IoT and Web 3.0. In *International Conference on Computer Applications in Industry and Engineering*; Springer Nature: Cham, Switzerland, 2024.
29. Fernández-Iglesias, M.J.; Delgado von Eitzen, C.; Anido-Rifón, L. Efficient Traceability Systems with Smart Contracts: Balancing on-chain and Off Chain Data Storage for Enhanced Scalability and Privacy. *Appl. Sci.* **2024**, *14*, 11078. [\[CrossRef\]](#)
30. Jadhav, S.; Pise, N. Secure and Transparent Blockchain Donations: An Attribute Based Access Control (ABAC) Framework. In *Proceedings of IEEE ICBDS*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–6.
31. Ullah, Z.; Husnain, G.; Mohmand, M.I.; Qadir, M.; Alzahrani, K.J.; Ghadi, Y.Y.; Alkahtani, H.K. Blockchain-IoT: A revolutionary model for secure data storage and fine-grained access control in IoT. *IET Commun.* **2024**, *18*, 1524–1540. [\[CrossRef\]](#)
32. Li, R.; Sun, Y.; Liu, C.; Wen, Y.; Liu, Y. Distributed Data Sharing and Access Control in Industrial IoT Using Blockchain Technology. In *IEEE ICCEIC*; IEEE: New York, NY, USA, 2024; pp. 372–375.
33. Cioloca, P.; Aloman, A.; Bratu, A.; Aciobăniței, I. Efficient Framework for Supply Chain Digitization using Blockchain and IoT. In *IEEE COMM*; IEEE: New York, NY, USA, 2024; pp. 1–6.
34. Akkal, M.; Cherbal, S.; Annane, B.; Lakhlef, H. Blockchain for Secure Access Control in the Internet of Things: A Review and Comparative Analysis. In *IEEE ICAECOT*; IEEE: New York, NY, USA, 2024; pp. 1–6.

35. Tuteja, S.; Sharma, M.; Sharma, A.; Tonk, R. Recent Emergence and Integration of Blockchain, Internet of Things (IoT), and Artificial Intelligence (AI) and their Future Directions. *Recent Adv. Electr. Electron. Eng.* **2025**, *18*, 1513–1526. [\[CrossRef\]](#)
36. Riaz Khan, M.; Abrar, N.; Tahir, M.R.; Mansoor, Y.; Rehman, S. Trustworthy AI for IoT using blockchain. *Spectr. Eng. Sci.* **2025**, *3*, 934–947.
37. Chandana, B.; Hussain, H.S.; Veluri, R.K.; Rajesh Kanna, S.K.; Babu, A.J.; Prasad, Y. Advancement of Smart Contract Administration in a Virtual Environment through Blockchain Technology and Optimization Tools. In Proceedings of the 3rd International Conference on Optimization Techniques in the Field of Engineering (ICOFE-2024), Online, 22–23 October 2024. [\[CrossRef\]](#)
38. Roobini, S.; Kavitha, M.S.; Karthik, S. Leveraging IoT and AI Technologies for Blockchain Based Digital Twins. In *Blockchain-Based Digital Twins*; Apple Academic Press: Palm Bay, FL, USA, 2025; pp. 183–222. [\[CrossRef\]](#)
39. Leghemo, I.M.; Azubuike, C.; Segun-Falade, O.D.; Odionu, C.S. Data Governance for Emerging Technologies: A Conceptual Framework for Managing Blockchain, IoT, and AI. *J. Eng. Res. Rep.* **2025**, *27*, 247–267. [\[CrossRef\]](#)
40. Anand, P.; Singh, Y.; Singh, H. Secure IoT data dissemination with blockchain and transfer learning techniques. *Sci. Rep.* **2025**, *15*, 1665. [\[CrossRef\]](#)
41. Nowell, E.; Gallus, S. Advancing Privacy Preserving AI: A Survey on Federated Learning and Its Applications. *Preprints* **2025**. [\[CrossRef\]](#)
42. Qiu, C.; Wu, Z.D.; Wang, H.; Yang, Q.; Wang, Y.; Su, C. Hierarchical Aggregation for Federated Learning in Heterogeneous IoT Scenarios: Enhancing Privacy and Communication Efficiency. *Future Internet* **2025**, *17*, 18. [\[CrossRef\]](#)
43. Nomikos, N.; Xilouris, G.; Patsourakis, G.; Nikolakakis, V.; Giannopoulos, A.; Mandilaris, C.; Gkonis, P.K.; Skianis, C.; Trakadas, P. A Distributed Trustable Framework for AI Aided Anomaly Detection. *Electronics* **2025**, *14*, 410. [\[CrossRef\]](#)
44. Murala, D.K.; Vemulapalli, L.; Balagoni, Y.; Patnala, E.; Romeo, B. MedLedgerFL: A hybrid blockchain-federated learning framework for secure remote healthcare services. *Sci. Rep.* **2026**, *16*, 8218. [\[CrossRef\]](#)
45. Yang, G.; Li, W.; Zhang, Q.; Xu, M.; Pan, C.; Wang, B. Trust-Enabled Decentralized Task Offloading for Collaborative Edge Computing Using Blockchain and Deep Reinforcement Learning. *Softw. Pract. Exp.* **2026**, *56*, 218–236. [\[CrossRef\]](#)
46. Galiveeti, P.; Pallavi, R. An internet of things edge intelligence powered by blockchain. In *Futuristic Trends in Blockchain Applications*; 2024; pp. 297–314. [\[CrossRef\]](#)
47. Oskolkov, B.; Chen, K.; Tian, W.; Law, A.C.C.; Liu, C. Incremental Machine Learning integrated Blockchain for Real time Security Protection in Cyber enabled Manufacturing Systems. *J. Comput. Inf. Sci. Eng.* **2025**, *25*, 041004. [\[CrossRef\]](#)
48. Norta, A.; Udokwu, C.; Voicu-Dorobanțu, R.; Ogunyemi, A.; Sturua, N.; Craș, S. Leveraging Blockchain for Ethical AI: Mitigating Digital Threats and Strengthening Societal Resilience. *Preprints* **2025**. [\[CrossRef\]](#)
49. Sabit, H. AI Based Smart Security System Using IoT for Smart Home Applications. *Preprints* **2025**. [\[CrossRef\]](#)
50. Deng, Y.; Guo, B.; Chen, S. Privacy Preserving Approach to Edge Federated Learning Based on Blockchain and Fully Homomorphic Encryption. *Electronics* **2025**, *14*, 361. [\[CrossRef\]](#)
51. Aounzou, Y.; Boulaalam, A.; Kalloubi, F. Convergence of blockchain, IoT, and machine learning: Exploring opportunities and challenges—A systematic review. *Int. J. Smart Sens. Intell. Syst.* **2025**, *18*. [\[CrossRef\]](#)
52. Perera, M.; Fattah, S.M.M.; Mistry, S.; Krishna, A. Reinforcement Learning Controlled Adaptive PSO for Task Offloading in IIoT Edge Computing. In *Companion Proceedings of the ACM on Web Conference 2025*; Association for Computing Machinery: New York, NY, USA, 2025. [\[CrossRef\]](#)
53. NISTIR 7628 Rev. 1; Guidelines for Smart Grid Cybersecurity. National Institute of Standards and Technology: Gaithersburg, MD, USA, 2014. [\[CrossRef\]](#)
54. IEC 62351:2026 SER; Power Systems Management and Associated Information Exchange—Data and Communications Security—All Parts. International Electrotechnical Commission: Geneva, Switzerland, 2026.
55. IEEE Std 2030.5-2023; IEEE Standard for Smart Energy Profile Application Protocol. Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2024.
56. NERC CIP Standards; Critical Infrastructure Protection Reliability Standards. North American Electric Reliability Corporation: Atlanta, GA, USA, 2026.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.