SN Computer Science (2025) 6:970
https://doi.org/10.1007/s42979-025-04519-8

ORIGINAL RESEARCH

Empowering Local Energy Communities with Blockchain-Based Federated Forecasting and Zero-Knowledge Proof Verification

Fabio Turazza[1] · Marcello Pietri[1] · Natalia Selini Hadjidimitriou[1] · Marco Picone[1] · Paolo Burgio[2] · Marco Mamei[1,3]

Received: 14 April 2025 / Accepted: 6 November 2025

Abstract

Local Energy Communities (LECs) are gaining prominence as key actors in the transition toward sustainable and decentralized energy systems. A critical challenge for these communities lies in achieving energy self-sufficiency through effective forecasting of energy production and consumption. Accurate forecasting models are essential to support optimization and planning strategies. However, privacy concerns and regulatory constraints often limit the feasibility of centralized data-driven approaches, as users are understandably reluctant to share their consumption data. To address this issue, we propose a privacy-preserving forecasting framework based on Federated Learning (FL) and Long Short-Term Memory (LSTM) networks, which enables collaborative model training without disclosing raw user data. Building upon this core architecture, we further enhance transparency and user engagement by introducing Zero-Knowledge Proofs (ZKPs) for secure inference verification, and a novel incentive layer based on dynamic Non-Fungible Tokens (dNFTs) and fungibile tokens. Our approach ensures model integrity, protects user data, and fosters sustainable behavior through verifiable, trustless reward mechanisms. Experimental results demonstrate the feasibility and potential of this architecture in supporting privacy-aware, decentralized energy forecasting within LECs.

Keywords Local energy communities (LECs) · Federated learning (FL) · Internet of things (IoT) · Zero-knowledge proofs (ZKPs) · Dynamic NFTs (dNFTs) · Blockchain

Introduction

Urban environments are increasingly at the forefront of addressing sustainable development and energy resilience, serving as practical testbeds for decentralised and participatory energy models due to their density of infrastructure and demand [1, 2]. In this context, cities and local communities naturally emerge as primary arenas to re-design production, distribution and consumption through decentralised and participatory arrangements.

Within the EU policy framework, Local Energy Communities (LECs) are recognised as citizen-led legal entities that coordinate local generation, storage and sharing. Member States are mandated to enable participation in renewable production, consumption, storage and trade, in line with the UN Agenda 2030 and the provisions of the Electricity Market Directive, the Renewable Energy Directive and the "2019 Clean Energy for All Europeans" Package [3–6].

While LECs offer numerous benefits, they also present challenges, particularly in community management due to rapidly changing regulations and uncertainties on how to

✉ Fabio Turazza
fabio.turazza@unimore.it

Marcello Pietri
marcello.pietri@unimore.it

Natalia Selini Hadjidimitriou
selini@unimore.it

Marco Picone
marco.picone@unimore.it

Paolo Burgio
paolo.burgio@unimore.it

Marco Mamei
marco.mamei@unimore.it

[1] DISMI, University of Modena and Reggio Emilia, Reggio Emilia, Italy

[2] FIM, University of Modena and Reggio Emilia, Modena, Italy

[3] En&Tech, University of Modena and Reggio Emilia, Reggio Emilia, Italy

Published online: 18 November 2025

enhance self-consumption and design the community effectively [7]. Figure 1 presents the LEC architecture: multiple users with diverse energy profiles share local resources within a community. The goal is to minimize reliance on the power grid by using and sharing local batteries and production.

In recent years, initiatives related to Energy Communities integrating smart grid technologies have increased [8, 9]. Smart grids enable bidirectional energy flows, demand management and flexible responses to overload. In systems powered by Renewable Energy Sources (RES), consumers can also supply energy back to the grid. These prosumers enable improved demand management, especially with the growth of electric vehicle (EV) charging. The spread of electromobility increases electricity demand, notably in high-density areas such as shopping centers, workplaces and airports. Within this framework, LECs offer a practical way to manage the rising EV charging demand.

Accurate short-term forecasting and continuous monitoring are essential to coordinate storage, allocate surplus, and plan EV charging, yet centralized data collection is often constrained by privacy and regulation [7]. Participants also resist sharing fine-grained load profiles, further limiting centralization.

In LECs, Federated Learning (FL) addresses privacy in energy forecasting by training robust models without sharing sensitive personal data. Rather than centralizing data, users collaboratively train a model while keeping data local. This approach meets growing data protection requirements such as GDPR and CCPA.

In this study, we apply a Federated LSTM (Long Short-Term Memory) combined with Ethereum-based components to produce accurate energy consumption forecasts while ensuring transparency and reliability. We show how these techniques can be integrated to preserve user privacy while providing actionable insights for energy management. Additionally, after training, we employ a customized zero-knowledge proof (ZKP) to track an individual user's energy profile via dynamic non-fungible tokens (dNFT), incentivizing reduced energy waste while maintaining strict privacy across training, inference and profiling. Leveraging FL and ZKP enables privacy-preserving ML that improves the efficiency and sustainability of LECs.

We also present an empirical case study based on an innovative dataset [10] to validate these ideas.

This paper is structured as follows: "Related Work and Contribution" section reviews the state of the art in LEC management and FL-based forecasting. "Architecture and Methodology" section details the proposed architecture, which integrates FL with Blockchain, ZKPs and dNFTs. "Experimental Setup andEvaluation" section presents the experimental setup, including ZKP performance evaluation, and the results obtained. Finally, "Conclusion" section discusses the main findings, limitations, and outlines directions for future work.

Related Work and Contribution

In recent years, the development of digital infrastructures for managing interactions within LECs has garnered significant attention as a key enabler of sustainable cyber-physical energy systems. These infrastructures provide the foundation for implementing forecasting and optimization mechanisms in LECs. Due to the inherent complexity of smart grids, hierarchical control structures have been widely

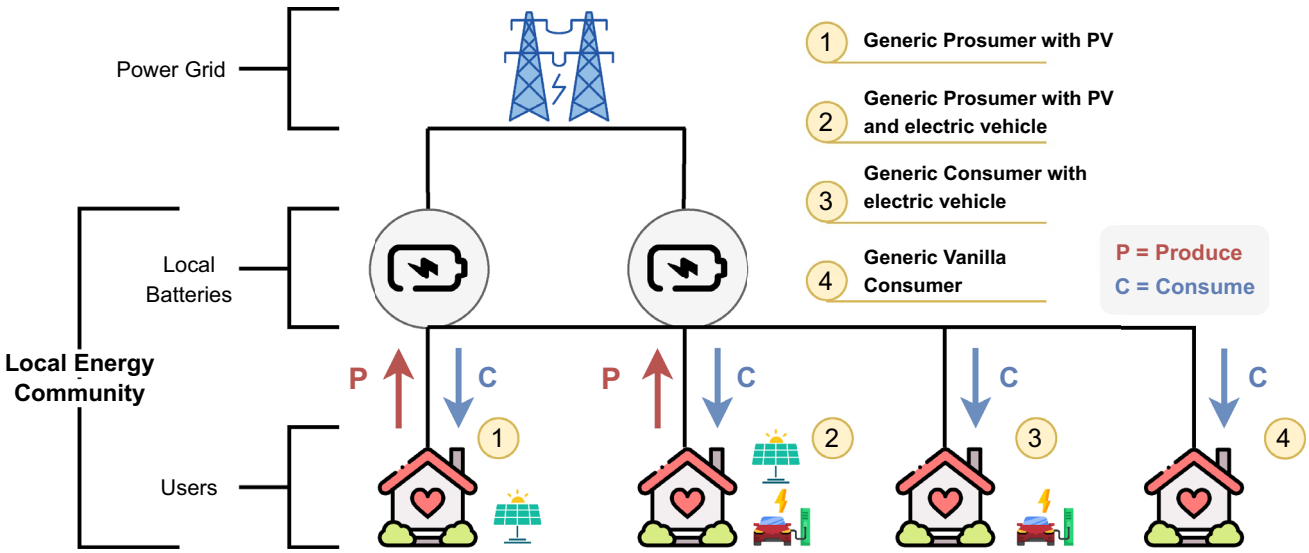

Fig. 1 Schematic of a local energy community (LEC): users with different energy profiles share local resources to reduce grid reliance via local batteries and production

explored. For example, [11] proposed a model optimizing the payoff between market actors and distribution operators. Similarly, [12] focused on Electric Vehicle (EV) charging coordination within microgrids, and [13] designed a hierarchical Digital Twin for smart grid simulation. In [14], a Community Energy System (CES)-based energy management scheme was introduced, enabling individual buildings to optimize their costs while the community storage system coordinates energy redistribution.

The concept of Peer-to-Peer (P2P) energy trading, which empowers prosumers to participate in the commercialization of surplus energy, has emerged as a catalyst for renewable energy adoption [15]. However, the real-time management of these exchanges requires reliable digital platforms and advanced metering infrastructures [16]. Moreover, the increasing penetration of EVs and energy storage systems further emphasizes the need for dynamic, decentralized energy optimization strategies.

Our forecasting model, based on FL, builds upon these digital infrastructures to enable energy exchange among prosumers equipped with storage and EVs, as well as conventional consumers. FL-based prediction provides a privacy-preserving approach to managing distributed energy data in LECs, enabling forecasting mechanisms essential for optimizing local energy flows.

Traditional centralized ML models aggregate datasets at a single point, exposing them to potential data breaches [17]. In contrast, FL, originally proposed by [18], enables decentralized model training without exposing raw data. This aspect is particularly relevant in energy contexts where data is inherently cross-silo, distributed among diverse entities, and privacy-sensitive. Nonetheless, FL presents challenges such as handling data heterogeneity, asynchronous updates, and communication overhead. Aggregation techniques like FedProx and FedMA [19] have been introduced to address these issues, while edge-based strategies have been proposed to reduce network strain [20].

Renewable energy domains such as solar and wind require models that are both adaptive and resilient to fluctuations. In such contexts, FL ensures privacy-preserving learning while addressing the inherent instability of energy data. However, the reluctance of customers to share consumption data, due to privacy concerns, commercial sensitivities, and cyber risks, remains a major barrier to the adoption of centralized models [19, 21].

While several works have explored forecasting within LECs, many such as [22] rely on centralized frameworks and omit FL, potentially compromising both scalability and privacy. FL, by enabling decentralized model training and coordination via a central aggregator, retains data locality and lowers communication overhead. Though still emerging in the energy domain, FL has seen successful applications in adjacent fields such as Internet of Things (IoT) and edge computing, offering strong privacy guarantees [19].

For instance, [23] designed an FL-Edge approach using LSTM networks for short-term residential load forecasting, improving performance through user clustering. Lin et al. [21] proposed an FL-based Bayesian Neural Network to disaggregate community-level solar generation while preserving privacy, using probabilistic modeling to capture uncertainties.

While existing works such as FederatedGrids [24] have demonstrated the potential of combining FL and Blockchain to enable decentralized energy trading and sharing, they focus mainly on the integration of prediction models within smart contracts to automate energy exchange among microgrids. Similarly, recent surveys on Blockchain-enabled Federated Learning (BCFL) [25] emphasize the role of blockchain in improving model traceability, attack resistance, and decentralized trust management, including privacy-preserving techniques such as differential privacy and homomorphic encryption. However, these approaches either neglect the inference phase or lack a fine-grained privacy-preserving verification mechanism that supports user-level auditability and incentives.

Complementary strands specifically target community-scale forecasting and coordination. Savi and Olivadese [23] propose an edge-centric FL with LSTM for short-term residential load forecasting, showing gains under data locality and user heterogeneity. Lin et al. [21] develop a privacy-preserving FL method for community-level "behind-the-meter" solar disaggregation with probabilistic uncertainty modelling. Bouachir et al. [26] (*FederatedGrids*) combine FL and blockchain to enable traceable P2P energy sharing across microgrids. Zhang et al. [27] investigate privacy-preserving FL for load forecasting with model visibility to aid interpretability. Taken together, these lines establish key building blocks for decentralised, privacy-aware analytics in LECs, while leaving open fine-grained, inference-time verification and user-level incentive design; the brief comparison in Table 1 situates these contributions.

Our work builds upon these contributions by applying FL within a comprehensive energy management framework tailored for LECs. Leveraging a novel synthetic dataset [10], we implement advanced LSTM-based forecasting while introducing aggregation strategies that improve performance across diverse user profiles. All model training occurs in a decentralized manner, ensuring strong data privacy guarantees.

Contributions

This work advances our previous contribution [29] and the existing literature, introducing three main innovations:

Table 1 Comparison with representative state-of-the-art (SoA) works; arrows indicate our incremental contributions

Some of SoA methods	Our key addition
Savi and Olivadese (2021) [23]	FL–edge LSTM for residential forecasting → add ZKP-verified inference and privacy-preserving per-user scoring.
Lin et al. (2022) [21]	Privacy-preserving FL for community solar disaggregation → extend to surplus forecasting across mixed user classes + on-chain proofs and incentives.
Bouachir et al. (2022) [26]	FL+Blockchain-assisted P2P energy sharing → introduce ZKP-anchored verification and dNFT-based incentives.
Zhang et al. (2022) [27]	Privacy-preserving FL for load forecasting with visibility → add on-device ZKPs and tokenised incentives.
Sameera et al. (2024) [25]	Survey on privacy in blockchain-based FL → operationalise with Groth16 proofs and verifiable scoring in LECs.
Groth (2016) [28]	Succinct pairing-based SNARK (Groth16) → tailor to federated LSTM updates with on-chain verification.

Zero-Knowledge Proofs at inference level: We propose the use of Zero-Knowledge Proofs (ZKPs) during the inference phase of the FL model. This allows users to prove, without disclosing sensitive data, that their actual energy consumption lies below a predicted threshold. The mechanism preserves privacy not only during training but also in the operational phase, addressing a key gap in existing BCFL frameworks.

On-chain verification via smart contracts: The ZKP is verified on a public blockchain by a smart contract, enabling a decentralized and trustless mechanism for validating user behavior. This ensures data integrity and model authenticity, addressing concerns in the literature regarding guarantees about the honesty of the global model shared in FL systems [25].

Dynamic NFTs as incentive layer: We introduce a gamified reputation and incentive mechanism based on dynamic NFTs (dNFTs) that evolve with user performance. These NFTs reflect each prosumer's efficiency over time and can unlock token-based rewards or access tiers. Unlike reward-driven BCFL methods relying on static token distribution [24], our approach supports long-term engagement through verifiable, privacy-preserving proofs.

This FL-ZKP-Blockchain architecture provides a unified framework that enhances privacy and auditability while introducing a transparent, incentive-aligned mechanism for energy efficiency in LECs. It addresses open challenges highlighted in recent surveys such as secure inference, model verifiability and incentive schemes and paves the way for privacy-centric, decentralized energy ecosystems.

Architecture and Methodology

Our approach consists of generating a dynamic energy profile of the user without sharing their private data. In the first stage, an LSTM model is trained in a federated manner to produce an annual forecast of the energy surplus of a community (aggregating predictions from individual users), enabling logistical estimations related to battery replacement and periods of energy overproduction. Each training step is traceable and certified through blockchain technology. In the second stage, the same federated model is used during inference to compare predictions with the user's actual consumption via a zero-knowledge protocol. This comparison generates a score, which is used to create a dynamic, On-Chain, NFT profile of the user, incentivizing reduced consumption behaviors.

LSTM Federated Training

As described in [30], FL takes into account N participants, i.e. members of the LEC that want to leverage trained model. Users merge their knowledges $\{p_1, p_2, \ldots, p_N\}$ to train the model. The FL model is trained by minimizing the loss function in Eq. (1). Where n_k is the amount of data on the member k and $F_k(w)$ is the local objective function.

$$\min f(w) = \sum_{k=1}^{N} \frac{n_k}{n} F_k(w) \tag{1}$$

The global model parameters are set up in a central server. Each client download the global parameters, trains its own model, and finally updates the global model round after round [31]. FL can be divided in three primary categorizations: horizontal FL, vertical FL, and Federated transfer learning. Horizontal type deals with overlapping features among the datasets of its members. In contrast, vertical one deals with overlapping users with less overlapping features. The last type rarely has overlapping datasets but it can overcome data scarcity by transferring learning [32].

In this work we focus on horizontal FL as we have multiple members of the LEC having datasets exhibiting overlapping features (i.e., energy consumption and production patterns for each member). To make predictions in forecasting, we employed a LSTM network, given its ability to model the multiple time-scales patterns associated to energy consumption and production [33]. Combining LSTM networks with FL offers a powerful approach for handling sequential data while preserving data privacy with some key advantages such as the reduced risk of overfitting with the aggregation of updates from multiple clients with potentially diverse datasets and the reduced communication costs

by processing data locally on devices, sharing only model updates, making the FL suitable for real-time predictions, even though this aspect will not be considered in this paper.

The method we propose to address this scenario is a Model-Centric Cross-Silo Horizontal architecture including several entities that represent individual prosumer or consumer users. These entities perform Client-Side Forecasting of the net energy production on an hourly basis.

$$Net\_Energy_x = \sum_x E_{exp,x} - \sum_x E_{imp,x} \quad (2)$$

- $E_{exp,x}$ = Total Net Energy from User x
- $E_{exp,x}$ = Exported_energy from User x
- $E_{imp,x}$ = Imported_energy from User x
- x = User Index (Prosumer or Consumer)

Subsequently, the parameters of the models containing the temporal relationships with a 24-hour dependency are aggregated Server-Side through a custom extension of Federated Averaging (FedAvg). This effectively creates a new dataset of parameters which, while maintaining the temporal relationships of the different clients, will be sent back to each client for a pre-defined number of federated epochs. At the end of federated learning, the predictions are centrally associated with the reference time instant through a FedProx aggregation algorithm and they are summed by filtering the frames for each index. In this way, we move away from the concept of individual user energy to talk about the net energy of the LEC. A synthetic architecture overview of the federated flow is shown in Fig. 2.

Blockchain Integration in Federated Settings

The integration of blockchain technology into FL settings significantly enhances transparency, security, and trustworthiness of the collaborative training processes. By leveraging Ethereum blockchain, each step in the federated training procedure, such as model updates, aggregation results, and training epochs, is immutably recorded and timestamped through Smart Contracts, enabling decentralized verification and traceability.

Furthermore, utilizing IPFS (InterPlanetary File System) in combination with Ethereum greatly enhances the system's scalability and efficiency. IPFS is a distributed, peer-to-peer storage system that efficiently stores data off-chain, providing content-addressable storage identified through cryptographic hashes [34]. This structure ensures data immutability, prevents redundancy, and facilitates secure and rapid retrieval of training artifacts like global model states and cryptographic proofs. The synergy between Ethereum's transparent ledger and IPFS's decentralized storage enhances system robustness by ensuring immutability and verifiability of model artifacts through on-chain hashes, while reducing costs by storing large files (e.g. models, weights, ZKP proofs) off-chain. This separation significantly lowers gas fees-the transaction costs users pay on a blockchain network to compensate validators for processing and securing transactions- and storage overhead, and improves overall reliability and accessibility of federated learning assets across distributed nodes.

Zero-Knowledge Proof of Correct Computation

ZKPs are cryptographic protocols that allow a prover to demonstrate to a verifier that a given computation was performed correctly, without revealing any of the underlying input data. This property is particularly valuable in privacy-sensitive

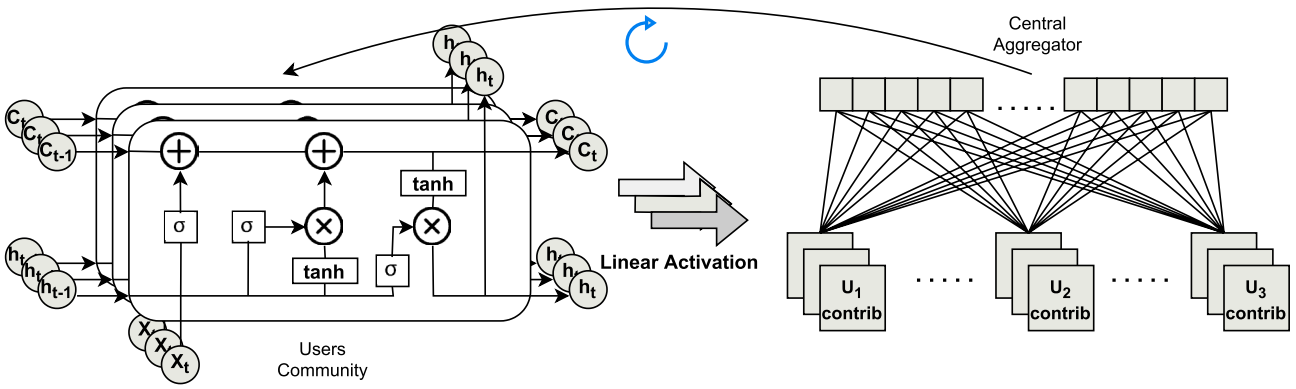

Fig. 2 Our proposed FL architecture: on the left, there are several LSTM layers that represent different client instances, also called 'runners'. The clients train on their local data using a model shared by the server and send the parameters to it. The server then acts as a central aggregator and combines the parameters to obtain a better global model

contexts, such as LECs, where verifying user behavior must not compromise individual confidentiality.

In our system, ZKP (specifically the Groth16 protocol [28]) is used to ensure the correctness of the inference process executed locally by each user.

In this context, a user wants to *prove* to the system that the inference to forecast his/her energy consumption has been executed correctly, and to compute the *user score*: a quantitative indicator comparing the user's actual energy consumption and the forecast provided by the federated model (see "Phase 2: Model Inference andProfiling through ZKP" section). This comparison is performed locally and verified through ZKPs, ensuring both correctness and privacy.

In extreme synthesis, a user obtains an un-modifiable software component (a *Circom circuit* [28]) able to cryptographycally sign (e.g., via Poseidon hash) and certify the inference and the resulting *user score*. The user runs inference via this *Circom circuit* and the resulting proof and *user score* are stored off-chain via IPFS and verified on-chain through smart contracts deployed on the Ethereum blockchain. This mechanism enables third parties, such as community managers or incentive systems, to validate user-reported results without accessing any raw consumption data or internal model parameters. As a result, our approach effectively balances transparency and confidentiality, strengthening user trust while enabling verifiable, privacy-preserving reward assignment within the federated ecosystem (Fig. 3).

dNFT Profilation

Dynamic Non-Fungible Tokens (dNFTs) are an evolution of traditional NFTs that allow on-chain metadata to change over time in response to external events or verified interactions. In our system, dNFT serve as evolving digital profiles that reflect the energy-related behavior of each user within a LEC. Each user is assigned a dNFT that encapsulates his/her *user score*, as described before, that reflects how well the user's behavior aligns with efficiency goals, without revealing any raw data. These dNFTs are deployed on the Ethereum blockchain and dynamically updated based on verified user performance. Their evolving nature enables the representation of behavioral trends over time, such as consistent energy savings or improvements in self-consumption. As such, dNFTs can function as decentralized and tamper-proof indicators of user reputation, potentially influencing access to community-based incentives, gamified reward schemes, or participation in higher-tier programs.

By transparently encoding sustainability achievements into a verifiable digital identity, dNFTs support both individual behavioral change and the collective goals of the LEC. They incentivize continuous engagement, foster accountability, and enable personalized incentive structures while preserving the privacy and autonomy of participants.

Design Considerations for Privacy and Incentive Alignment

The integration of ZKPs and dNFTs within the proposed FL framework addresses the dual objective of preserving user privacy and enabling transparent, verifiable incentive mechanisms in LECs.

Unlike centralized forecasting systems that rely on absolute accuracy, our approach adopts a relative assessment strategy. Instead of requiring perfect predictions, the system performs a secure comparison, validated via ZKPs, between predicted and actual consumption values. This comparison yields a user-specific performance score, which is evaluated against dynamic thresholds derived from statistical benchmarks or personal consumption history. In this way, incentives are awarded based on improvements relative to the user's own past behavior or to the average of their user category (e.g., prosumers vs. consumers), effectively

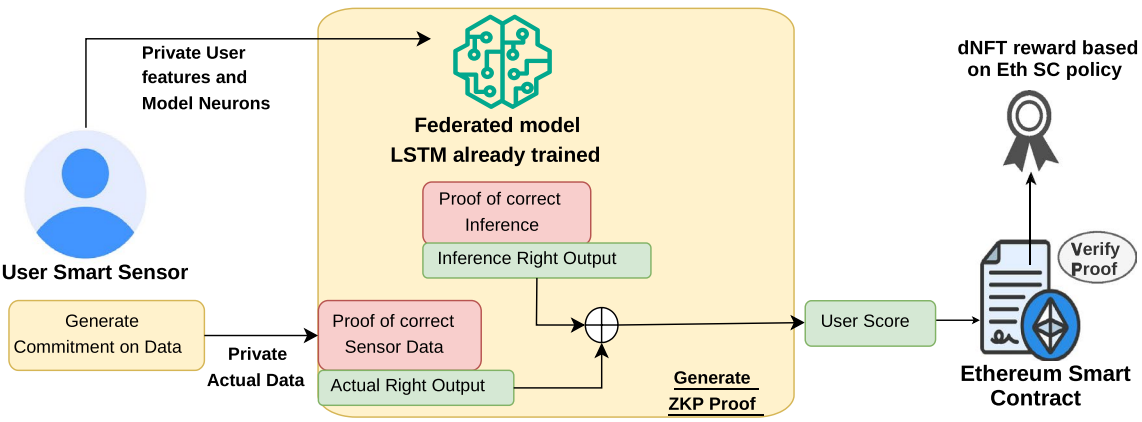

Fig. 3 Overview of the privacy-preserving inference workflow. The user commits private sensor data, runs local inference with a federated LSTM, and generates a ZKP. The Ethereum smart contract verifies the proof and updates the user's dNFT based on the score and reward policy

mitigating the bias introduced by structural heterogeneity among participants.

A critical challenge in this architecture concerns the risk of users manipulating the data used to generate proofs, for instance, by falsifying consumption values to obtain rewards. To address this, the system assumes the use of trusted data acquisition devices, such as certified smart meters capable of generating cryptographic commitments at the source. ZKPs are then computed from these committed values and validated on-chain via smart contracts. To further strengthen the robustness of the system, optional mechanisms such as randomized challenge-response protocols or delegated third-party auditing can be incorporated to detect anomalies or inconsistencies in reported data.

Another key aspect is the trustworthiness of the federated model used during inference. In traditional FL settings, clients cannot verify that the global model they use corresponds to the one they contributed to during training. To resolve this, we adopt a model notarization mechanism whereby each aggregated global model version is hashed and its digest recorded on-chain. This provides an immutable and verifiable reference, enabling any user to validate that the inference model is authentic and unaltered, thereby reinforcing both model integrity and accountability.

Temporal dynamics of energy behavior also inform the incentive structure. Short-term prediction windows, daily or weekly, can offer fast feedback and immediate engagement, while longer-term evaluations, monthly or seasonal, are better suited for measuring consistent improvement and resilience. The system supports both, enabling layered incentive schemes verified independently via ZKP mechanisms.

The introduction of dNFTs enhances the incentive system by adding a persistent and reputational dimension. Each user is assigned a unique dNFT that evolves over time according to their verified performance. These tokens encapsulate key behavioral metrics, including efficiency scores and milestone achievements, and serve as decentralized, tamper-resistant profiles. dNFTs can unlock community privileges, tiered benefits, or be integrated into tokenized ecosystems (e.g., via ERC-20 compatibility), thus supporting both engagement and reward personalization in a privacy-preserving and trustless manner.

The proposed design addresses several open challenges in decentralized energy systems, including user privacy preservation across all stages (training, inference, reward), protection against manipulation, verifiable trust in model updates, and alignment of individual behavior with collective sustainability goals through multi-scale, transparent incentive structures.

Theoretical Underpinnings

FL–LSTM (goal & gist). We learn a shared model from non-IID clients by minimizing $f(\mathrm{w}) = \sum_{i=1}^{N} p_i \, \mathbb{E}_{x\sim\mathcal{D}_i}[\ell(\mathrm{w}; x)]$, $\sum_i p_i = 1$. Assume L-smooth local losses and bounded heterogeneity $\delta = \frac{1}{N}\sum_i \|\nabla f_i(\mathrm{w}^\star)\|^2$. Standard results give convergence to an ε-stationary point with rate $\mathcal{O}(1/T)$ plus a non-IID drift term $\mathcal{O}(\eta E^2 \delta)$ (learning rate η, E local steps). *Interpretation:* the second term explains accuracy loss under skewed data and motivates FedProx (penalize drift), balanced client sampling, and modest E. For fixed-point quantization q, each multiply adds $\leq 2^{-q}$ error; an LSTM cell with $\kappa \approx 4(HI + H^2)$ multiplies yields per-time-step bias $\mathcal{O}(\kappa\, 2^{-q})$.

ZKP (what we prove & cost). Public inputs: commitment $\mathrm{cm}(y)$ to smart-meter readings, model hash $h(\mathrm{w})$, and round id; witness (y, w). We prove that the forecast $\hat{y}$ computed from w is consistent with y and achieves loss $\leq \tau$ without revealing y. With a succinct SNARK (e.g., Groth16), proof size and verification time are (near) constant, while prover time scales as $T_P = \alpha\, N_{\mathrm{constr}}$ with $N_{\mathrm{constr}} = \mathcal{O}(\kappa T)$ for sequence length T.

dNFT incentives (why they align behavior). Using a strictly proper scoring rule $S(y, \hat{y})$ makes truthful forecasting optimal. We mint a dNFT when $\mathrm{Verify}(\pi) = \top$ and $S(y, \hat{y}) \geq \tau$, embedding round hash, score, and commitments, yielding per-user auditability, privacy (no raw data on-chain), and Sybil resistance (one identity, one reward).

End-to-end scaling.

$$T_{\mathrm{round}} \approx \mathcal{O}\big(C\,|\mathrm{w}|\big) + \alpha\, N_{\mathrm{constr}} + \mathcal{O}(1) + g_{\mathrm{mint}} \,.$$

Here C is the active-client fraction, $|\mathrm{w}|$ model size, and g_{mint} the on-chain minting cost.

Why these technologies. FedProx reduces non-IID drift without heavy coordination; *Groth16*-class SNARKs give constant-size proofs and fast on-chain verification; *dNFTs* (vs. fungible tokens) bind rewards to user/round/score for transparent, non-fungible auditability; *on-device proofs* at the smart meter ensure authenticity without exposing raw measurements.

Experimental Setup and Evaluation

In this study, we conducted a series of experiments focused on predicting electricity surplus availability within a LEC. Our aim is to assess a range of LEC configurations, including varying percentage of prosumers and consumers in order to study recurring patterns within these communities.

We built a 400-user dataset (200 prosumers, 200 consumers) from [10] by concatenating days and applying an exponentially weighted moving average (window = 120 h). This is a dataset created by using real consumption data of Danish residents and consists of hourly energy production and consumption on weekends/holidays and weekdays in the four seasons.

Given this dataset, we applied our LSTM-FL to forecast energy surplus availability of multiple prosumers within a LEC. The LSTM is configured as stateless and receives as input the features from a single user such as historical energy flows in a year, time of the day, and temperature. Furthermore, the model generates a 24-hour energy forecast for each user. The training process learns the surplus behaviour from multiple similar users within the LEC. Once the network is trained, we will be able to predict the energy output generated by each user for each period of the year. Summing all the predictions, filtered by year and time slot, it allows us to derive the energy surplus available to the LEC. The following experiments were all conducted on a laptop equipped with a 13th-generation Intel i7 processor, 32 GB RAM, and an NVIDIA RTX 4080 GPU laptop.

To ensure privacy and verifiability, the inference results are validated using ZKPs (Groth16), allowing each user to prove correctness of their predicted output without revealing underlying data. These proofs are stored off-chain and verified on-chain via smart contracts, enabling secure and trustless surplus reporting.

Phase 1: Federated Training Evaluation

In our experimental evaluation, the first analysis focuses on surplus energy prediction across the three distinct scenarios, including a Stand-Alone model (thus considering single-user communities), a Centralized model trained on a dataset formed by concatenating all the raw data from a LEC, and finally, a Federated model trained on the same data as the Centralized model but without the explicit sharing of data within the LEC. The following line-charts (Fig. 4) show the Mean Squared Error (MSE) loss obtained in the different approaches considering a small subset of 10 users. In Fig. 4-right, we can observe the actual homogeneity in the training loss of the various clients in the Federated approach. Every LSTM model was trained using a 3-layers LSTM network with a hidden layer of 50 neurons, we used Adam as optimizer algorithm. The models were trained on 8 features:

- `day` (`int`): The day of the month, represented as an integer (0–335).
- `index` (`int`): The daily hours, represented as an integer (0–24).
- `temperature_median` (`float`): The median temperature recorded, represented as a floating-point number.
- `temperature_std` (`float`): The standard deviation of the temperature, represented as a floating-point number.
- `season` (`str`): The season of the year (e.g., winter, spring), represented as a string.
- `type` (`Bool`): A boolean value indicating the type of User (Prosumer = 0, Consumer = 1).
- `type2` (`int`): An additional type represented as an integer indicating a subset type of Users and what kind of energy they produce and consume.
- `sum_total_ewm_balance` (`float - target`): The total sum of the balance per hour (energy surplus), represented as a floating-point number.

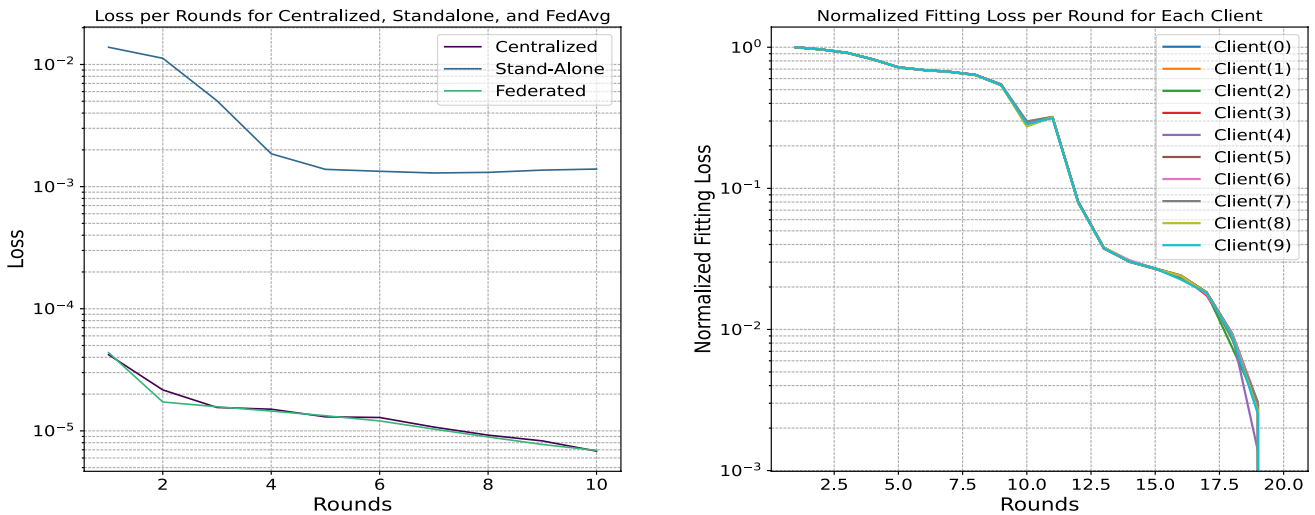

Fig. 4 Left: MSE of the three approaches (Stand-Alone, Federated, Centralized) on a small subset of 10 users. Right: 10-clients LEC MSE fitting losses performance comparison over time

As illustrated in Fig. 4-left, It is evident that performance improves drastically when moving from a Stand-Alone architecture to a Centralized or Federated one. Between these latter two architectures, we naturally observe slightly better performance in the Centralized model compared to the Federated one, which, however, preserves the privacy of each individual user's data.

For the Federated experiment, the performance was compared using five of the most well-known different aggregation methods (shown in Fig. 5) including FedAvg, FedMedian, FedProx, FedAdam and FedYogi.

FedProx [35] was chosen for this experiment due to its effectiveness in handling non-IID (Non-Independent and Identically Distributed) data, which is common in FLs-cenarios. Unlike FedAvg (that combines the local model updates from multiple clients by simply averaging them), which assumes that all clients' data is identically distributed, FedProx, or Federated Proximal, introduces a proximal term ($\mu = 0.01$) to the traditional framework. This term is designed to tackle the challenges posed by heterogeneous data distributions among users. By penalizing deviations of local updates from the global model, FedProx maintains stability and accelerates convergence ensuring that the local updates remain closer to the global model, which enhances the efficiency of the training process and contributes to faster convergence.

The objective function of the FedProx algorithm in the context of FL is defined as follows:

$$\min_{w} \left\{ \frac{1}{K} \sum_{k=1}^{K} \left(\frac{1}{n_k} \sum_{i=1}^{n_k} f_{k,i}(w) \right) + \frac{\mu}{2} \|w - w^t\|^2 \right\} \tag{3}$$

Where:

- $\min_w$ indicates that we are minimizing with respect to the global weight vector w.
- $\frac{1}{K} \sum_{k=1}^{K}$ represents the average over all clients.
- $\frac{1}{n_k} \sum_{i=1}^{n_k} f_{k,i}(w)$ represents the average of the losses over the samples from each client k.
- $\frac{\mu}{2} \|w - w^t\|^2$ is the Prox regularization term, which penalizes deviations from the global weights w^t of the current iteration.

The term $\frac{1}{K} \sum_{k=1}^{K} \left(\frac{1}{n_k} \sum_{i=1}^{n_k} f_{k,i}(w) \right)$ represents the average loss over data from different clients, while the term $\frac{\mu}{2} \|w - w^t\|^2$ acts as a Prox regularization term that helps stabilize the optimization by penalizing deviations from the previous iteration's global weights.

To increase the security of the system while maintaining an adequate level of performance without overloading it with significant computational overhead, global differential privacy (DP) was applied to the system's aggregated model, along with gradient clipping on the model (which helps to bound the sensitivity of the gradients and prevent excessive updates). We adopt global DP on aggregated updates to bound sensitivity and limit accuracy loss compared to local DP.

Methodology All experimental parameters and assumptions are explicitly reported in "Experimental Setup andEvaluation" and "Phase 1: Federated TrainingEvaluation" sections: model (three-layer LSTM, hidden=50, linear head), optimization (Adam, lr=1×10^{-3}, $\beta = (0.9, 0.999)$, batch=64), window/horizon (24 h/24 h), feature set (day, hour, temperature_median/std, season, type, type2), target scaling (min–max [0,1]), user-stratified split (70/15/15), per-client normalization (z-score), FL setup (10 rounds, 1 local epoch/round, client fraction $C = 0.2$), aggregation (FedProx with $\mu = 0.01$; comparisons with FedAvg/

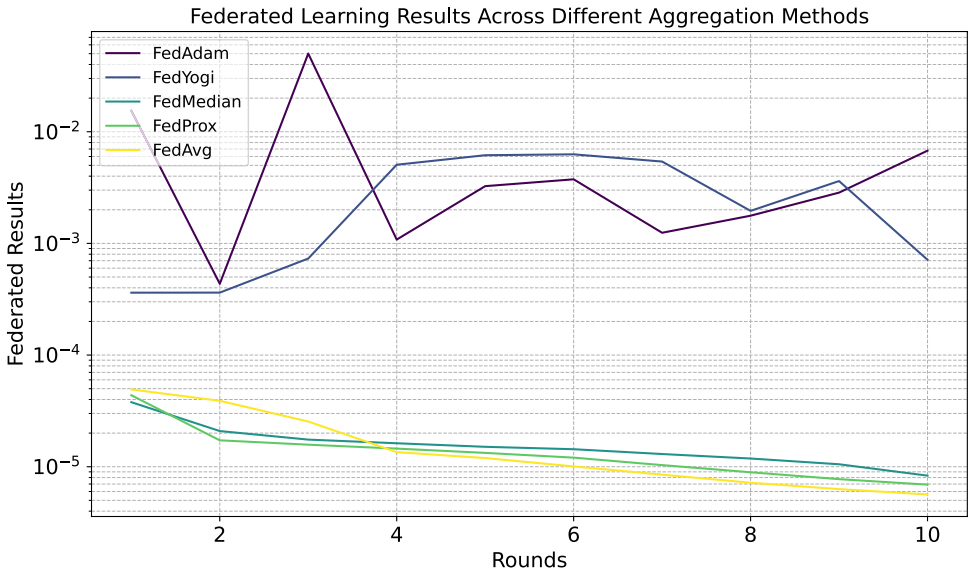

Fig. 5 Federated aggregation methods MSE loss comparison including five of the most used aggregation algorithms on a 10-users small subset

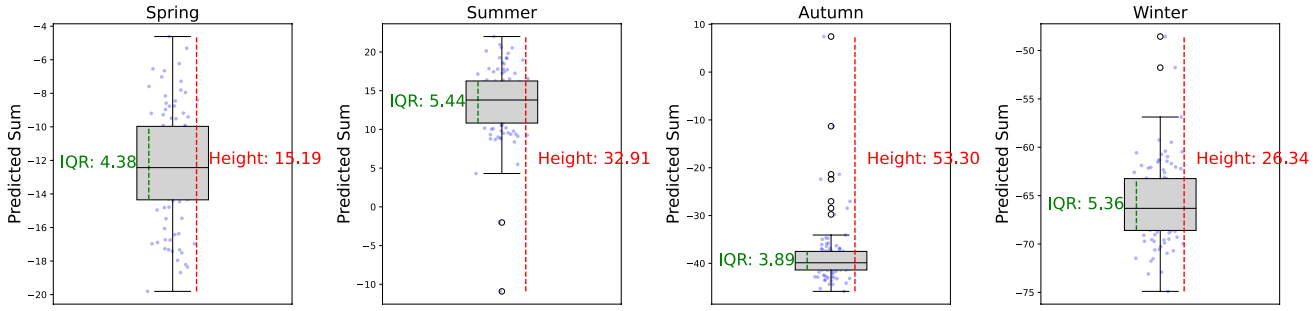

Fig. 6 Seasonal box-plot which shows the distribution of energy consumptions of a 50/50 LEC over different seasons

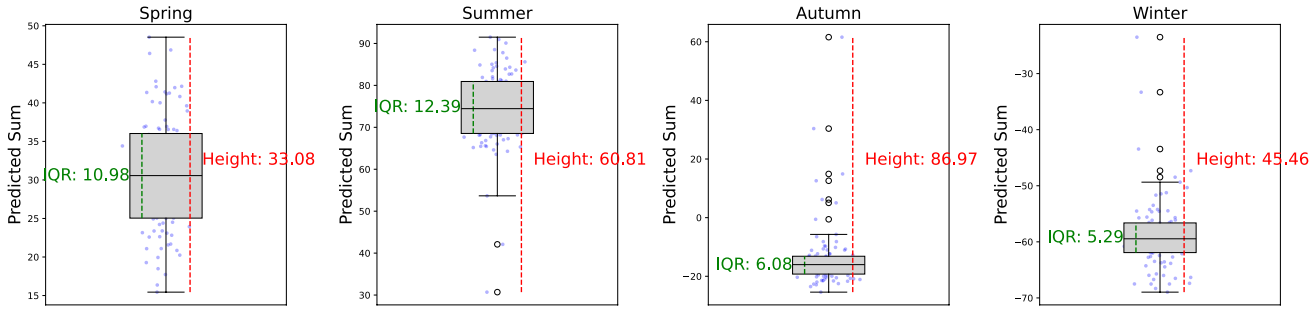

Fig. 7 Seasonal box-plot which shows the distribution of energy consumptions of a full prosumers LEC (FPL) over different seasons

Median/Adam/Yogi), and hardware (i7-13th, 32 GB RAM, RTX 4080 Laptop GPU). This paragraph serves as a compact confirmation: no additional assumptions beyond those listed are used.

Mixed LEC Community

In a mixed LEC, prosumers produce and supply excess energy to consumers. In PV-based local energy communities (LECs), overproduction, defined as the surplus of generated solar power beyond local consumption, often occurs in spring and early summer, when solar irradiance peaks while demand remains comparatively low. Note that communities dominated by other renewables (e.g., wind-based LECs) may exhibit different seasonal overproduction patterns. The impact of consumers in a community (mixed LEC) is primarily observed during warm seasons, a period in which the variability (measured by the difference between the third quartile ($Q3$) and the first quartile ($Q1$)) among inter-LEC values increases by more than 150% in the spring months, reaching an increase of only 1.32% in the winter values. This data makes forecasting the warm seasons more uncertain, encompassing a much larger range of values. This trend is not reflected uniformly in the length of the whiskers in the box-plot, which remains fairly consistent across seasons, excluding any potential outliers. The interquartile range metrics through box-plots were obtained by comparing a homogeneous mixed LEC (Fig. 6) (50% prosumers, 50% consumers) with a Full Prosumers LEC (FPL) (Fig. 7).

In the line chart above (as shown in Fig. 8), the actual final prediction of the Federated model after a 10-rounds session in a mixed community is represented. The model was trained on a dataset of 257 users, with hourly tracking, evenly split between prosumers and consumers. The same model was tested on an equivalent dataset of 100 users, divided into 50% prosumers and 50% consumers, with the number decreasing as the percentage of consumers declines. For clarity, all users in our experiments are residential; we do not include commercial or industrial consumers. Each residential user is further classified into one of five types such as PV-only prosumers, PV + ESS prosumers, PV + EV prosumers, EV-only consumers, and vanilla consumers.

- `Type 0`: Prosumer with a photovoltaic (PV) system, $\sim$ 60 *Users considered*.
- `Type 1`: Prosumer with a PV system and an energy storage system (ESS), $\sim$ 59 *Users considered*.
- `Type 2`: Prosumer with a PV system integrated with an electric vehicle (EV), $\sim$ 58 *Users considered*.
- `Type 3`: Consumer with an electric vehicle (EV) without PV integration, $\sim$ 89 *Users considered*.
- `Type 4`: Consumer, labeled as "Vanilla" represents a standard or baseline consumer type with typical energy consumption, not associated with specialized systems like EVs or PV systems, $\sim$ 91 *Users considered*.

Note that the dataset was trained and tested on a small community with only four generation types; wind power was not

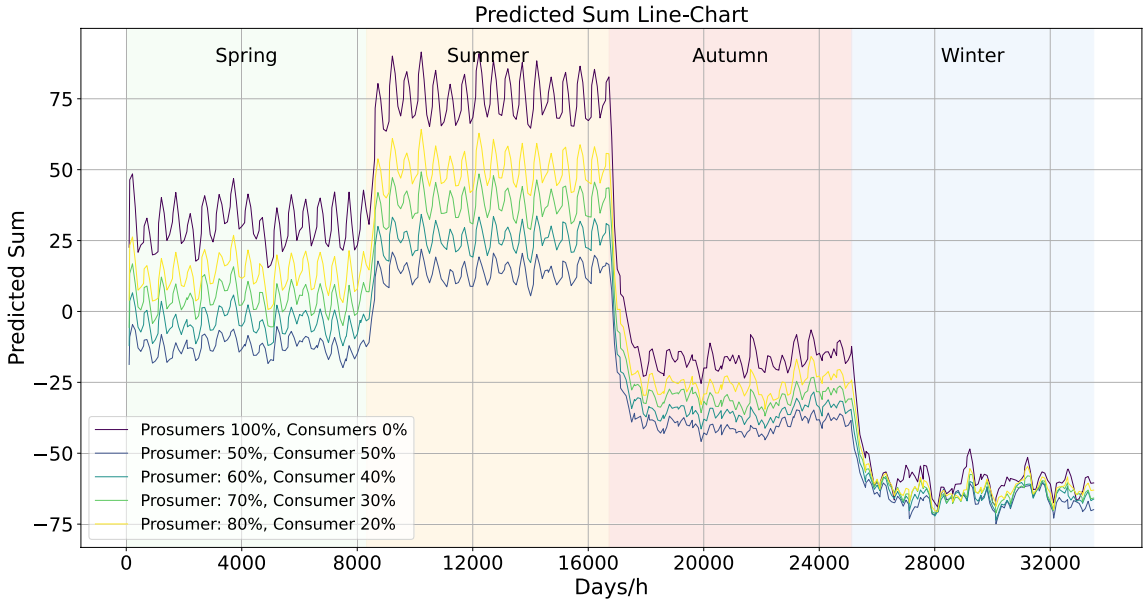

Fig. 8 Comparison between different mixed communities, divided by percentages of consumers. The following values are derived from tests conducted in homogeneous communities (i.e., communities with similar user type distributions), each consisting of 100 users within a LEC

included. This can bias the evaluation in larger communities where the wind, Denmark's second most adopted source after photovoltaics, plays an important role, especially in winter and spring when the wind peaks while PV is scarce. The second bias is due to climate: the data reflect Danish conditions (cold winters and milder springs / summers compared to southern Europe), which limits generalization to hotter regions with greater temperature variability.

Therefore, the final model forecasts the energy surplus accumulated by the community for each hour of the year (for visualization purposes, a value every 404 h is shown in the line chart). The ideal value is obtained by subtracting the imported energy from the produced energy after performing an exponentially weighted mean of every imported/exported form of energy. The simulation, therefore, provides a value for each hour of the year based on the composition of the LEC, indicating the current energy availability. This value can be positive in the case of a surplus of produced energy, which is a trend observed during the summer and spring months, and negative in the case of a surplus of consumed energy, despite the produced energy, which is a trend observed during the autumn and winter months.

As shown in Table 2, the model achieves the lowest MSE on EV-only consumers (0.0097) and the highest on vanilla consumers (0.0132). This suggests that load patterns with well-defined charging cycles are easier to predict than uncorrelated, vanilla profiles. Overall, the small variance across categories indicates that our federated LSTM handles heterogeneous user behaviors robustly.

Table 2 Forecasting errors by user type: mean squared error (MSE), mean absolute error (MAE), and root-mean-square error (RMSE)

User type	MSE	MAE	RMSE
PV-only prosumers	0.0123	0.088	0.111
PV + ESS prosumers	0.0108	0.083	0.104
PV + EV prosumers	0.0115	0.086	0.107
EV-only consumers	0.0097	0.078	0.098
Vanilla consumers	0.0132	0.094	0.115

Phase 2: Model Inference and Profiling Through ZKP

In our system, we integrate the Groth16 Zero-Knowledge Proof (ZKP) protocol [28] to securely verify federated inference computations while maintaining complete confidentiality of user data. Groth16 operates through an initial trusted setup phase (4), conducted offline, which generates essential cryptographic parameters: a proving key and a corresponding verifying key. These keys enable efficient proof generation and succinct verification of computations:

$$(\mathrm{pk}, \mathrm{vk}) \leftarrow \mathrm{Setup}(1^{\lambda}, \mathcal{C}) \tag{4}$$

where pk = proving key, vk = verifying key, 1^{λ} = security parameter, and $\mathcal{C}$ is the arithmetic circuit we named *UnifiedInferenceComparison*.

For commitment verification, the circuit employs the Poseidon hash function:

$$C = \mathrm{PoseidonHash}(y_{\mathrm{predicted}}) \tag{5}$$

$$= f_{\text{Poseidon}}\left(\sum_{i=1}^{n} x_i \cdot \alpha_i + c \mod p\right) \tag{6}$$

where C is the cryptographic commitment, f_{Poseidon} is the Poseidon hash function, α_i are Poseidon constants, and p is a large prime number.

Within the unified circuit, the actual value (provided along with its secret nonce and a public commitment) is verified via the Poseidon hash before it is used to compare with the model's prediction. The final user score is computed as follows:

$$\text{Score}_{\text{user}} = \begin{cases} 0, & \text{if } d = 0, \\ \frac{d}{T}, & \text{if } |d| \le T, \\ \frac{d}{T} + \text{sgn}(d)\,\alpha\left(\frac{|d|-T}{T}\right), & \text{if } |d| > T \end{cases} \tag{7}$$

where $d = y_{\text{actual}} - y_{\text{predicted}}$, T is the deviation threshold, and α is a scaling parameter.

To mitigate bias and ensure consistency across users, the model is retrained to perform weekly forecasts; daily forecasts could lead to unfair evaluations by excessively penalizing users with occasional consumption spikes.

Proofs generated via Groth16 consist of compact cryptographic artifacts and related public outputs. Formally, the proof generation for our circuit is expressed as follows:

$$w_{\text{UIC}} \leftarrow \text{WitnessGen}_{\mathcal{C}_{\text{UIC}}}(x_{\text{private}}, x_{\text{public}}) \tag{8}$$

$$\pi_{\text{UIC}} \leftarrow \text{Prove}(\text{pk}, w_{\text{UIC}}) \tag{9}$$

where w_{UIC} is the witness[1], π_{UIC} is the Groth16 proof, x_{private} denotes private inputs, and x_{public} denotes public inputs.

To optimize blockchain resource efficiency, these proofs and outputs are securely stored off-chain using IPFS (InterPlanetary File System), leveraging its content-addressable, decentralized storage capabilities. The combination of Ethereum and IPFS ensures data immutability, efficient retrieval, reduced redundancy, and lower on-chain storage costs. In our project, the entire ZKP lifecycle and the on-chain verification phase is illustrated on the Prover Side (Users) in Fig. 9.

We specifically selected Groth16 due to its distinct advantages over alternative zero-knowledge proof protocols [28]. Compared to other protocols such as Bulletproofs or STARKs, Groth16 proofs offer smaller sizes and faster verification speeds; qualities critical within the cost-sensitive Ethereum blockchain environment. The succinctness of Groth16 proofs significantly reduces transaction costs (gas fees) and computational overhead during on-chain verification. Although Groth16 requires an initial trusted setup phase, this structured process provides enhanced security assurances beneficial in our federated scenario, ensuring that no sensitive user information is exposed.

The on-chain verification step conducted by our Ethereum smart contracts (see Fig. 9) can be formally represented as:

$$\{0, 1\} \leftarrow \text{Verify}(\text{vk}, x_{\text{public}}, \pi_{\text{UIC}}) \tag{10}$$

where vk is the verifying key, x_{public} = the public inputs, π_{UIC} is the Groth16 proof generated by the circuit *UnifiedInferenceComparison* (5), and the verification output is 1 (valid) or 0 (invalid).

[1] Note on the Witness: The witness (8) is a collection of all the values computed within the circuit that satisfy its constraints. It is crucial for generating a valid proof because it encapsulates the confidential data needed to demonstrate that the computation was computed correctly without revealing any information.

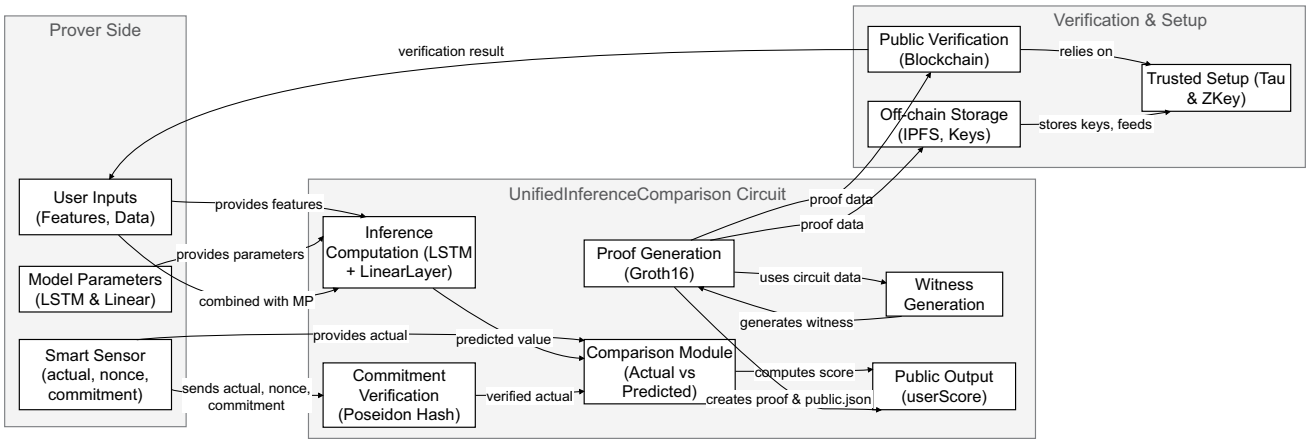

Fig. 9 This diagram illustrates our offline trusted setup phase, where cryptographic keys are generated, followed by Groth16 proof generation. It involves a unified circuit that handles private inputs (features, model parameters, the actual value and a threshold), computes predictions and verifies the commitment of the actual value, and compares it with the prediction to produce a public user score

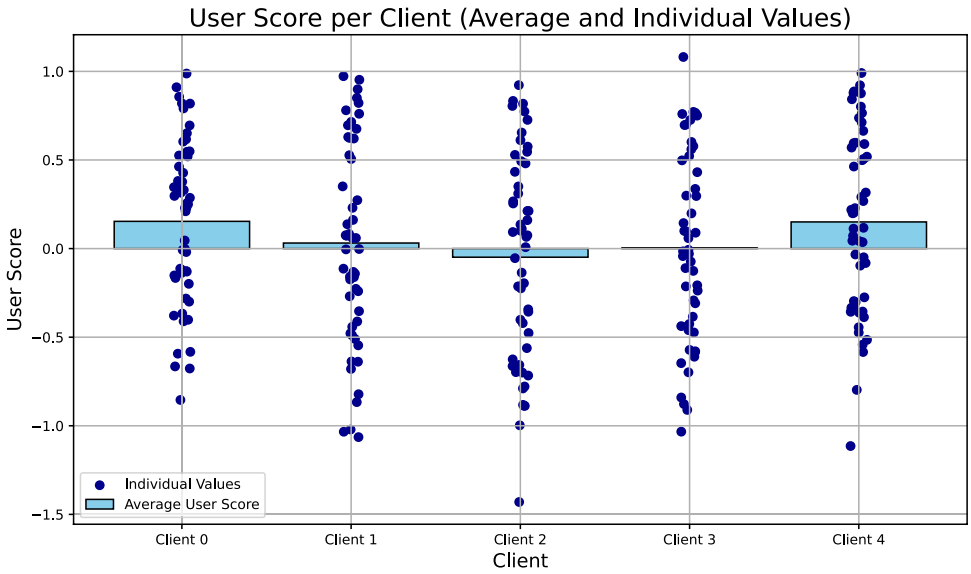

Fig. 10 This bar chart shows the user score obtained by a subset of five users over a 52-week period. The dots represent individual weekly updates. The score (7) is continuously updated and reflects the user's energy efficiency: a higher score indicates better performance

Table 3 zkSNARK on-chain costs and circuit metrics with peer-reviewed references

Module	Measured	Evidence from literature
(a) On-chain costs (Ethereum L1)		
Groth16 verify	$t_{\text{verify}} \approx 1.4$ s	<230k gas to verify on Ethereum L1 in zkBridge evaluations [36, 37]
Groth16 prove	$t_P \approx 5.0$ s	Off-chain; time scales with circuit constraints (SNARK theory) [28]
Proof size	–	Constant-size proof (3 group elements) for Groth16 [28]
PLONK verify	–	Constant-size proof; efficient verifier on EVM reported in practice; see protocol properties [38]
ERC-721 mint	$t_{\text{mint}} \approx 0.1$ s	Empirical studies report tens–low-hundreds k gas per mint, depending on implementation [39, 40]
(b) Circuit metrics: UnifiedInferenceComparison		
Template instances	81	
Non-linear constraints	217,879	
Linear constraints	2780	
Public inputs	0	
Private inputs	43,945	
Public outputs	1	
Wires	264,603	
Labels	966,347	
Proving key size	∼115.04 MB	

(a) We placed side by side our measured times with evidence on Ethereum verification/mint costs: Groth16 verification <230k gas on L1 in zkBridge [36] and NDSS follow-up [37]; Groth16 proofs are constant-size [28]; PLONK offers constant-size proofs with efficient verification [38]; ERC-721 mint costs measured in IEEE ICBC and SpringerOpen case studies [39, 40]. (b) Circuit metrics for `UnifiedInferenceComparison` in our setup.

Integrating zero-knowledge proofs like Groth16 delivers substantial benefits within our system, notably rigorous privacy protection, verified computational correctness, and strengthened user trust. Users and external parties can reliably confirm the integrity of inference results without ever compromising user privacy, thereby maintaining an ideal balance between transparency and confidentiality. Consequently, our adoption of Groth16 significantly enhances privacy, integrity, and scalability of our federated blockchain-based system, directly contributing to its practical viability and user adoption.

Based on Table 3-a, we can see that our measured Groth16 verification time ($t_{\text{verify}} \approx 1.4$ s) is consistent with the <230 k gas on Ethereum L1 as reported by zkBridge and an NDSS follow-up, placing us in *low end* of the widely reported SNARK verification costs on L1 [36, 37]. In contrast, STARK verifiers are commonly in the multi-million gas range, so the total on-chain envelope for our design, i.e., *verify* + *mint* (∼200–230k + 70–150k gas), remains $>\mathbf{10}\times$ cheaper than STARK-based approaches while being comparable to or below typical PLONK verifier ranges [38]. In addition, Groth16's constant-size proof (≈192 B) minimizes calldata overhead [28]. The verification is off-chain and scales with circuit constraints, making it the only tunable bottleneck (mitigated via client sampling and parallel provers). In general, the ZKP + dNFT layer exhibits a footprint on the chain *practical* on L1 / L2 and positions our implementation at the most cost-effective end of SNARK-based designs [28, 36–40].

Interpretation of key results. Across our experiments, federated forecasting attains accuracy close to the centralized baseline while preserving per-user privacy (cf. Figs. 4, 5), with FedProx (μ=0.01) showing the most stable convergence under non-IID users and adaptive rules (FedAdam/

FedYogi) reducing early-round variance. Error patterns align with user behavior and seasonality: EV-focused users exhibit the lowest MSE, in agreement with regular charging routines, while generic consumers remain harder to forecast due to greater behavioral variance; warmer months expand the dispersion in mixed communities, in line with the higher volatility of photovoltaic generation (Figs. 6, 7, Table 2).

On the trust and accountability layer, the blockchain ledger records commitments and scores without materially affecting model quality, while adding auditability to the incentive pipeline. Groth16-based ZK proofs validate per-user inference and target compliance with compact sizes and verification times adequate for periodic attestations (Table 3 and Fig. 11); in this regime the end-to-end overhead remains compatible with community-scale deployments. Overall, results indicate that (i) FL provides privacy-preserving forecasting with a modest gap to centralized training, (ii) the on-chain incentive/recording stack ensures verifiable accountability and (iii) ZK proofs make individual compliance attestable without revealing private load profiles.

Limitations

It's important to emphasize that managing biases is extremely relevant in a realistic context such as that of energy communities. In the examples provided earlier, regarding the inference phase, calculating a user score and subsequently assigning a dynamic NFT only makes sense if the comparison is made among users of the same category (prosumers and consumers should not have the same score). In a real mixed LEC scenario, the two categories should be further differentiated with distinct rewards and penalties.

Another practical limitation of the system concerns the need for a smart meter or, more generally, a certified device that can generate a commitment on the sensor-side data. Therefore, to ensure complete security and authenticity of the data at the source and to guarantee that a user cannot cheat the process by providing fake data, the system cannot be stand-alone.

In the Table 3-b, we present our ZKP system metrics, where one can see the number of constraints for each circuit and the proving key sizes. As we can observe, the proof takes a very high number of private inputs in order to replicate the model's structure and constrain the computation at each layer, so as to generate a correct proof of inference. Finally, a proof of correct computation approach is used to compare the model's output with the actual private value. The proof generates only one validated public output representing the user score, which is saved on the blockchain in the form of a reward token.

As can be seen from the diagrams in Fig. 11, the true advantage offered by Groth16 lies in the limited size of the zkp proof, with a longer verification time compared to other approaches. However, the longer verification time is not a crucial feature in this system, given that our objective is to conduct weekly proofs.

External Validity and Scale

Scope of our feasibility claim. Our synthetic LEC ($N \approx 400$ users) is smaller than many real deployments (1k–10k meters). We therefore qualify feasibility as *prototype-scale*: communication, proving and on-chain verification behaved as expected at $N \sim 10^2 - 10^3$ and we analytically project to larger N below.

Measured instantiation. From Fig. 11 we take Groth16 prover/verification times of $t_P \approx 5.0 \pm 0.6$s and $t_{\text{verify}} \approx 1.4 \pm 0.2$s; Table 3 reports a proving key of ~ 115 MB and $\sim 2.18 \times 10^5$ non-linear constraints. Our 3-layer LSTM (H=50, 8 features) has $\sim 5.4 \times 10^4$ parameters, i.e., $|\mathrm{w}| \approx 0.216$MB in float32.

Per-round projection (plugging measured values). Using Eq. (11) with conservative WAN rates (uplink 10MB/s, downlink 40MB/s) and moderate client sampling C=0.2:

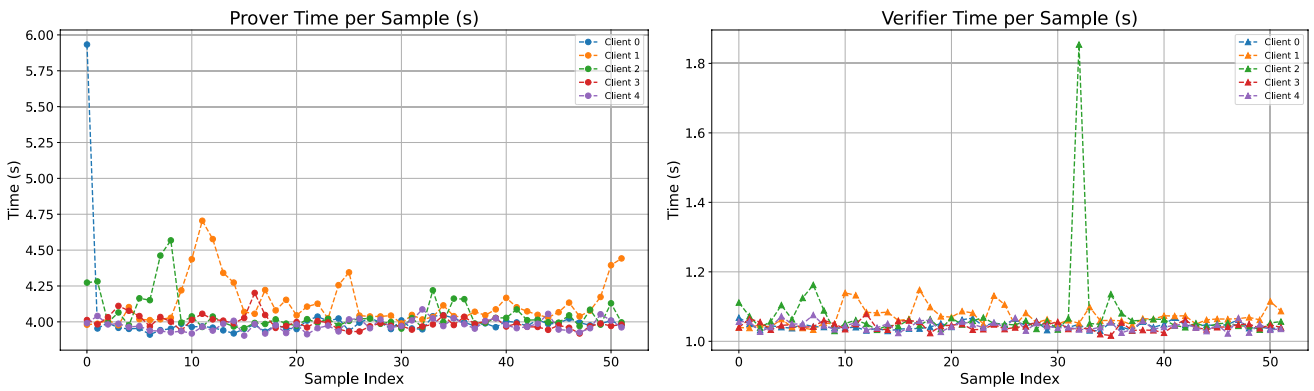

Fig. 11 Left: the prover-side time required to generate the ZKP proof for each client, Right: the verifier-side time required to verify the proof in our simulated environment

Table 4 Per-round time from Eq. (11) using measured $t_P \approx 5.0$s, $t_{\text{verify}} \approx 1.4$s, $|w| \approx 0.216$MB; $C=0.2$, $P \in \{32, 32, 64\}$, conservative WAN

	N = 400	N = 1000	N = 5000
$T_{\text{round}}(N)$ [s]	14.0	32.8	79.6

$$T_{\text{round}}(N) = \max\left\{ \frac{CN}{P} t_P,\ \frac{CN\,|\mathrm{w}|}{B_\uparrow} + \frac{CN\,|\mathrm{w}|}{B_\downarrow} \right\} + t_{\text{verify}} + t_{\text{mint}}\,. \tag{11}$$

Instantiating with $|\mathrm{w}| = 0.216$MB (WAN rates are conservative; with $|\mathrm{w}|$ this small the network term remains subdominant), $t_P = 5.0$s, $t_{\text{verify}} \approx 1.4$s, $t_{\text{mint}} \approx 0.1$s:

$$\begin{aligned} N{=}400,\ P{=}32: &\quad T_{\text{round}} \approx \mathbf{14.0}\ \text{s},\\ N{=}1000,\ P{=}32: &\quad T_{\text{round}} \approx \mathbf{32.8}\ \text{s},\\ N{=}5000,\ P{=}64: &\quad T_{\text{round}} \approx \mathbf{79.6}\ \text{s}. \end{aligned}$$

Here the network term stays <3s at $N=400$ and <28s at $N=5000$; the bottleneck is the prover. Increasing P (parallel provers) or lowering C (client sampling) reduces T_{round} linearly in the proving-dominated regime. With measured t_P, t_{verify}, and $|\mathrm{w}|$, the framework remains practical at $10^3 - 10^4$ users under moderate C and substation-level parallelism P, while on-chain verification adds a near-constant tail.

Dataset representativeness. To mitigate synthetic–real mismatch, we (a) report class proportions and non-IID skew used to generate users, (b) include distributional checks (e.g., KS/MMD on PV/ESS/EV marginals vs. public LEC statistics), and (c) note that FL accuracy degrades if classes are under-represented; hence new LECs should ensure near-IID sampling or balanced participation.

Scalability considerations. We demonstrate *prototype-scale feasibility* and provide closed-form scaling laws indicating that, under moderate client sampling and parallel proving, the framework remains practical for LECs with 10^3–10^4 participants (Table 4).

Conclusion

The application of federated learning (FL) to LECs enables decentralized, privacy-preserving forecasting by collaboratively training models on localized data. We further integrate zero-knowledge proofs (ZKPs) to validate inference correctness and provide verifiable compliance with forecast targets and we couple this with a dual-layer incentive scheme (fungible tokens + dNFTs) for per-user, auditable rewards.

Correct scoring and incentives must reflect structural differences (e.g., prosumers vs. consumers) to mitigate bias in reward distribution. Our evaluation shows that Groth16-based proofs are compact and verifiable with acceptable overhead, supporting practical deployment, provided that certified smart meters can generate tamper-resistant commitments at the edge.

To generalize our framework to other LECs, the following key requirements should be met:

Incentive rationale	justify an immutable, privacy-preserving scoring infrastructure despite its added complexity.
On-device proof	smart meters must generate initial ZKPs to prevent falsified data.
Rich sensing	deploy enough high-resolution data streams to train deep LSTM models.
IID distribution	ensure all user classes are well represented to avoid FL performance degradation.

Looking ahead, we will integrate the forecasting and incentive pipeline into real-time LEC optimization/control and explore edge computing, hardware enclaves, lighter ZK circuits and model compression to reduce latency and communication.

Author Contributions All authors contributed to the manuscript.

Funding Open access funding provided by Università degli Studi di Modena e Reggio Emilia within the CRUI-CARE Agreement. Work supported by NextGenerationEU project: Ecosystem for Sustainable Transition in Emilia-Romagna (Ecosister) CUP: B33D21019790006 - PNRR - Missione 4 Componente 2 Investimento 1.5 - Spoke 4. This research has also been supported by the project A catalyst for EuropeaN ClOUd Services in the era of data spaces, high-performance and edge computing(NOUS), GA: 101135927.

Data Availability The data are openly available in [10].

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Human and Animal Rights Not applicable.

Informed Consent Not applicable.

References

1. Otamendi-Irizar I, Grijalba O, Arias A, Pennese C, Hernández R. How can local energy communities promote sustainable development in European cities? Energy Res Soc Sci. 2022;84:102363.
2. Ceglia F, Esposito P, Marrasso E, Sasso M. From smart energy community to smart energy municipalities: literature review, agendas and pathways. J Clean Prod. 2020;254:120118.
3. United Nations: transforming our world: the 2030 agenda for sustainable development. 2015.
4. Directorate-general for energy, European Commission: proposal for a directive of the European parliament and of the council on common rules for the internal market in electricity (recast) COM (2016)864, 24/02/2017. 2017.
5. European Parliament: directive (EU) 2018/2001 of the European Parliament and of the council, of December 11, 2018, on the promotion of energy from renewable sources. 2021.
6. Union E. Clean energy for all Europeans package 2019.
7. Manso-Burgos Á, Ribó-Pérez D, Gómez-Navarro T, Alcázar-Ortega M. Local energy communities modelling and optimisation considering storage, demand configuration and sharing strategies: A case study in valencia (spain). Energy Rep. 2022;8:10395–408.
8. Barbour E, Parra D, Awwad Z, González MC. Community energy storage: a smart choice for the smart grid? Appl Energy. 2018;212:489–97.
9. Summeren LF, Wieczorek AJ, Bombaerts GJ, Verbong GP. Community energy meets smart grids: reviewing goals, structure, and roles in virtual power plants in Ireland, Belgium and the Netherlands. Energy Res Soc Sci. 2020;63:101415.
10. Yuan R, Pourmousavi SA, Soong WL, Black AJ, Liisberg JA, Lemos-Vinasco J. A synthetic dataset of Danish residential electricity prosumers. Sci Data. 2023;10(1):371.
11. Manshadi SD, Khodayar ME. A hierarchical electricity market structure for the smart grid paradigm. IEEE Trans Smart Grid. 2015;7(4):1866–75.
12. Wu Y, Wang Z, Huangfu Y, Ravey A, Chrenko D, Gao F. Hierarchical operation of electric vehicle charging station in smart grid integration applications—an overview. Int J Electr Power Energy Syst. 2022;139:108005.
13. Jiang Z, Lv H, Li Y, Guo Y. A novel application architecture of digital twin in smart grid. J Amb Intell Humaniz Comput. 2022;13(8):3819–35.
14. Nagpal H, Avramidis I-I, Capitanescu F, Madureira AG. Local energy communities in service of sustainability and grid flexibility provision: hierarchical management of shared energy storage. IEEE Trans Sustain Energy. 2022;13(3):1523–35.
15. Soto EA, Bosman LB, Wollega E, Leon-Salas WD. Peer-to-peer energy trading: a review of the literature. Appl Energy. 2021;283:116268.
16. Piselli C, Salvadori G, Diciotti L, Fantozzi F, Pisello AL. Assessing users' willingness-to-engagement towards net zero energy communities in Italy. Renew Sustain Energy Rev. 2021;152:111627.
17. Zhang B, Tan WJ, Cai W, Zhang AN. Forecasting with visibility using privacy preserving federated learning. In: 2022 winter simulation conference (WSC), 2022; 2687–2698
18. McMahan HB, Moore E, Ramage D, Hampson S, Arcas BA. Communication-efficient learning of deep networks from decentralized data 2016.
19. Cheng X, Li C, Liu X. A review of federated learning in energy systems. In: 2022 IEEE/IAS industrial and commercial power system Asia (I &CPS Asia), 2022; 2089–2095.
20. Wang T, Liu Y, Zheng X, Dai H-N, Jia W, Xie M. Edge-based communication optimization for distributed federated learning. IEEE Trans Netw Sci Eng. 2022;9(4):2015–24.
21. Lin J, Ma J, Zhu J. A privacy-preserving federated learning method for probabilistic community-level behind-the-meter solar generation disaggregation. IEEE Trans Smart Grid. 2022;13(1):268–79.
22. Putz D, Gumhalter M, Auer H. The true value of a forecast: assessing the impact of accuracy on local energy communities. Sustain Energy Grids Netw. 2023;33:100983.
23. Savi M, Olivadese F. Short-term energy consumption forecasting at the edge: a federated learning approach. IEEE Access. 2021;9:95949–69.
24. Bouachir O, Aloqaily M, Özkasap Ö, Ali F. FederatedGrids: Federated learning and blockchain-assisted P2P energy sharing. IEEE Trans Green Commun Network. 2022.
25. Sameera KM, Nicolazzo S, Arazzi M, Nocera A, Rehiman R, Vinod P, Conti M. Privacy-preserving in blockchain-based federated learning systems. arXiv preprint arXiv:2401.03552 2024.
26. Bouachir O, Aloqaily M, Ozkasap O, Ali F. Federatedgrids: federated learning and blockchain-assisted p2p energy sharing. IEEE Trans Green Commun Network. 2022;6(1):424–36.
27. Zhang B, Tan WJ, Cai W, Zhang AN. Forecasting with visibility using privacy preserving federated learning. In: 2022 winter simulation conference (WSC), 2022; 2687–98.
28. Groth J. On the size of pairing-based non-interactive arguments. In: Annual international conference on the theory and applications of cryptographic techniques (EUROCRYPT), vol. 9665. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer; 2016; 305–26.
29. Turazza F, Pietri M, Hadjidimitriou NS, Mamei M. Forecasting energy availability in local energy communities via LSTM federated learning. In: Proceedings of the 16th international conference on management of digital ecosystems (MEDES 2024), Naples, Italy 2024.
30. Wen J, Zhang Z, Lan Y, Cui Z, Cai J, Zhang W. A survey on federated learning: challenges and applications. Int J Mach Learn Cybern. 2023;14(2):513–35.
31. Zheng G, Kong L, Brintrup A. Federated machine learning for privacy preserving, collective supply chain risk prediction. Int J Prod Res. 2023;61(23):8115–32.
32. Zhang C, Xie Y, Bai H, Yu B, Li W, Gao Y. A survey on federated learning. Knowl-Based Syst. 2021;216:106775.
33. Greff K, Srivastava RK, Koutník J, Steunebrink BR, Schmidhuber J. Lstm: a search space odyssey. IEEE Trans Neural Netw Learn Syst. 2016;28(10):2222–32.
34. Psaras Y, Dias D. The interplanetary file system and the filecoin network. In: 2020 50th annual IEEE-IFIP international conference on dependable systems and networks-supplemental volume (DSN-S) 2020.
35. Sahu AK, Li T, Sanjabi M, Zaheer M, Talwalkar A, Smith V. On the convergence of federated optimization in heterogeneous networks. arXiv. 1812.06127 2018.
36. Xie T, Zhang J, Cheng Z, Zhang F, Zhang Y, Jia Y, Boneh D, Song D. zkbridge: Trustless cross-chain bridges made practical. arXiv. 2210.00264 2022.
37. Scaffino G, Magazzeni D, Bartoletti M, Maffei M, Pietro RD, et al. Alba: the dawn of scalable bridges for blockchains. In: Network and distributed system security symposium (NDSS) 2025.
38. Gabizon A, Williamson ZJ, Ciobotaru O. PLONK: Permutations over Lagrange-bases for oecumenical noninteractive arguments of knowledge. IACR cryptology ePrint archive, Report 2019/953 2019.
39. Choi W, Woo J, Hong JW-K. Gas cost analysis of fractional nft on the ethereum blockchain. In: 2023 IEEE international conference on blockchain and cryptocurrency (ICBC) 2023.

40. Sharma M, Baid M, Gupta A, Kumar P. Patient-centric decentralizing health records: a gas-efficient soulbound token framework for ehr management. J Eng Appl Sci. 2025.