

# Trust-Aware Federated Learning for Robust and Secure Social Media Analysis

Shalini Singh <sup>1\*</sup>, Mridula Singh <sup>1</sup> and Manash Sarkar <sup>2</sup>

1 Computer Science and Engineering, Quantum University, Roorkee, Uttarakhand, 247667, India; mridula.cse@quantumeducation.in

2 Computer Science and Engineering, Haldia Institute of Technology, West Bengal, 721657 India; manash.sarkar26@gmail.com

\* Correspondence author: shalinisinghphd08@gmail.com

**Abstract:** The rapid growth of social media requires privacy-preserving approaches for detecting toxic, fraudulent, misleading, and inappropriate content while protecting sensitive user information. This paper proposes a trust-aware federated learning framework that enables collaborative model training without sharing raw user data. An attention-based Long Short-Term Memory (LSTM) network captures important semantic relationships in textual content, while a trust scoring mechanism identifies unreliable or potentially malicious clients. A blockchain layer provides secure verification of model updates and enhances transparency throughout the federated training process. Experimental results demonstrate strong classification performance with limited blockchain overhead, highlighting the framework's potential as a scalable, secure, and efficient solution for privacy-preserving social media content analysis.

**Keywords:** Federated Learning, Blockchain; LSTM; Attention Mechanism; Privacy Preservation; Trust Score; Social Media Security.

## 1. Introduction

The rapid expansion of social media platforms has transformed the way people communicate and express their opinions. However, the widespread adoption of these platforms has also resulted in the increased dissemination of malicious, unethical, and misleading content, including hate speech, fake news, and explicit material [1]. Consequently, there is an urgent need for effective and scalable content moderation techniques. Traditional content moderation approaches, which rely on manual annotation or centralized Machine Learning (ML) models, suffer from several limitations related to scalability, privacy, and transparency [2]. Furthermore, the collection and processing of users' personal data by centralized servers raise significant privacy concerns, while centralized moderation systems are susceptible to bias, censorship, and single points of failure [3].

To address these challenges, Federated Learning (FL) [4] has emerged as a decentralized learning paradigm in which multiple participants, such as social media servers, user devices, and regional content moderators, collaboratively train a shared deep learning model without exchanging their raw data. Instead, each participant trains the model locally and shares only the model parameters or gradients with the federation. As a result, Federated Learning enhances user privacy and facilitates compliance with the General Data Protection Regulation (GDPR). Despite these advantages, FL faces several challenges when applied to social media content moderation. One of the major concerns is the lack of trust among participating clients, as malicious or compromised participants may submit poisoned or corrupted model updates, thereby degrading the performance of the global model. Additionally, the presence of non-independent and identically distributed (non-IID) data across clients can negatively affect model convergence and prediction accuracy. Another limitation is the reliance on a central aggregation server, which introduces a potential single point of failure and may become a target for attacks.

The integration of Blockchain with Federated Learning has emerged as a promising solution to overcome these limitations by improving security, transparency, and trust. Blockchain provides an immutable distributed ledger that securely records and verifies all transactions, including model updates, without requiring a trusted third party. Furthermore, smart contracts can automate participant

verification, model aggregation, and incentive distribution, ensuring that only validated model updates contribute to the global model. This integration, commonly referred to as Blockchain-Enabled Federated Learning (BFL) [5], enhances accountability, tamper resistance, traceability, and decentralized trust management.

Nevertheless, not all participants in a BFL system contribute reliable model updates. Poor-quality updates may originate from participants with noisy local datasets, limited computational resources, or malicious intent, such as model poisoning attacks. To mitigate these issues, this work introduces a Trust Scoring mechanism that evaluates the reliability of each participant based on historical behaviour, similarity of model updates, and contribution quality. The computed trust scores are securely maintained on the blockchain and are incorporated into the aggregation process by assigning higher weights to trustworthy participants during global model construction.

In this paper, we propose a Blockchain-Enabled Federated Deep Learning Framework with Trust Scoring (BFL-TS) for privacy-preserving and trustworthy social media content moderation. The proposed framework combines the privacy-preserving capabilities of Federated Learning, the transparency and immutability of blockchain technology, and a trust-aware aggregation mechanism to ensure fair and reliable collaboration among distributed participants. An attention-based Long Short-Term Memory (LSTM) network is employed to learn semantic representations from social media content, enabling accurate classification of textual and visual data while preserving user privacy. The proposed trust-aware framework ensures that only credible participant contributions significantly influence the global model. The main contributions of this paper are as follows:

1. A novel Blockchain-Enabled Federated Learning framework that eliminates the need for a central aggregator by using blockchain smart contracts to manage and verify global model updates, ensuring decentralization and transparency.
2. A Trust Scoring Mechanism that dynamically evaluates the reliability of participants based on their historical performance, model update similarity, and contribution accuracy, mitigating the risks of data poisoning and model manipulation.
3. A Privacy-Preserving Deep Learning Approach for social media content moderation that enables collaborative training of high-performance models without compromising user data privacy.

The use of blockchain, federated learning, and trust-aware techniques ensures that the proposed method meets the urgent demand for content moderation on social media platforms. This is achieved through a scalable base for building Artificial Intelligence (AI)-based content moderation systems that adhere to ethical AI guidelines and data privacy laws.

## 2. Related Work

Social media has led to an explosion of misinformation, disinformation, and fake news, necessitating urgent research efforts to preserve information integrity. In this regard, Aïmeur et al. [1] investigated algorithmic approaches for detecting misinformation and fake news, and their results demonstrate the effectiveness of machine learning (ML) and Natural Language Processing (NLP)-based algorithms in identifying misleading information. In addition to technology-driven approaches for mitigating misinformation, issues related to policy and governance are equally crucial, as argued by Oyinloye et al. [3]. The researchers emphasize that appropriate policy measures should be enacted to address the adverse societal effects of information mismanagement. Furthermore, algorithmic bias, misogynistic behaviour, online abuse, and cyberbullying pose serious ethical challenges, and many scholars have proposed artificial intelligence (AI)-based approaches to address these issues. Abdullah et al. [6,7] and Mohasseb et al. [8], for instance, have explored the use of AI for addressing online harassment and cyberbullying.

Concurrently, FL has been proposed as an effective approach for decentralized machine learning, enabling collaboration among participating parties to develop models without directly exchanging the personal information of end users. Saif et al. [2] investigated the benefits of decentralization in machine learning systems, particularly in terms of privacy preservation and transparency. Wen et al. [4] reviewed existing FL architectures and their associated challenges, including communication overhead and system heterogeneity. To provide stronger privacy guarantees in federated learning systems, Liu et al. [9] introduced advanced techniques such as differential privacy and cryptography, while FedV [10] and SAFELearn [11] provided frameworks for implementing these approaches. Issues related to trust, fairness, and bias in FL systems have also been investigated in [12].

The integration of blockchain with federated learning has further enhanced the security, transparency, and auditability of AI systems operating in distributed environments. Bajwa et al. [5], for example, investigated blockchain-enabled FL techniques that improve data integrity and resistance to tampering. Meanwhile, Mahato et al. [13] presented a verifiable FL framework based on homomorphic encryption

to secure the model-update process. Similarly, Nguyen et al. [14] extended this concept to edge computing environments. Furthermore, Kalapaaking et al. [15] and Nazir et al. [16] proposed IoT-based systems incorporating FL techniques. Recent studies have also explored trust-based decentralized AI models to improve reliability and transparency within distributed learning frameworks. For instance, Kridera and Kanavos [17] and Ahmed et al. [18] investigated trust-based models that can enhance collaboration among distributed nodes. Meanwhile, Chen et al. [6] and Liu et al. [19] highlighted the importance of explainability and system dependability in AI-based decision-making processes.

Rawat and Rajavat [20] proposed a trust-aware federated user representation learning framework for privacy-preserving and carbon-optimized personalized recommendation in large-scale social platforms. The study integrates federated learning with trust management to protect user information while improving the personalization of recommendations. The incorporation of carbon-aware optimization also highlights the importance of energy-efficient and sustainable intelligent systems. However, the approach is primarily focused on recommendation systems rather than secure communication and intrusion detection.

Singh et al. [21] investigated secure data transactions in social networks by analyzing the influence of network members. Their work emphasizes the role of trusted participants and secure interactions in protecting data exchanged through social networking environments. The study demonstrates the significance of trust and member influence in improving data transaction security. Nevertheless, it does not extensively address intelligent attack detection or adaptive security mechanisms for identifying sophisticated cyber threats.

EL-Sayed et al. [22] introduced the Tweet Tone Triage (4T) technique, which employs secured federated deep learning for analyzing and classifying tweet tones while preserving user privacy. The framework demonstrates the applicability of federated deep learning to privacy-sensitive social media analytics and reduces the need to centrally collect user data. However, its primary objective is social-media text classification, leaving opportunities for further investigation of federated deep learning for real-time intrusion detection, secure network monitoring, and communication-network security.

These studies demonstrate the growing importance of federated learning, trust management, privacy preservation, and deep learning in secure intelligent systems. However, limited attention has been given to integrating these concepts for real-time and adaptive intrusion detection in next-generation communication networks, which motivates the development of more comprehensive intelligent security frameworks.

## *2.1 Research Gap*

However, despite the considerable advancements made, several research areas remain open. Current research on disinformation, misinformation, and social media trust focuses primarily on content-based analysis and user behaviour, with comparatively less attention given to privacy-preserving data sharing and governance. This limitation restricts the applicability of existing approaches in practical and sensitive scenarios.

Although FL [14] and decentralized machine learning [2] can provide enhanced privacy compared with centralized approaches, most existing FL applications [10–12] rely on structured and relatively homogeneous datasets. In addition, very few existing studies consider social media trust and related indicators when developing an FL model, which can significantly influence the model aggregation process. Existing blockchain-based federated learning approaches address various technical aspects of trust, including secure aggregation, data integrity, and traceability. However, these approaches often overlook social trust aspects, making the system transparent but not necessarily interpretable or explainable. Furthermore, existing studies on fairness, bias mitigation, and inclusivity have largely been conducted independently, without adequately considering decentralized and privacy-preserving approaches. Therefore, the absence of an integrated approach combining trust models, ethical governance, and privacy-oriented architecture represents a critical research gap. Consequently, there is a need to develop a comprehensive framework that integrates trust, privacy, and federated learning through blockchain technology.

## *2.2 Novel Contribution*

The proposed Trust-Aware Federated Learning framework is motivated by the limitations of existing centralized and conventional federated learning approaches in handling privacy, unreliable participants, malicious model updates, and data heterogeneity. While federated learning preserves data locality, it does not inherently ensure that all participating clients are trustworthy or that their updates are reliable. The proposed approach addresses these limitations by combining local model training with trust-aware aggregation and blockchain-based verification, thereby protecting raw social media data while reducing

the influence of unreliable clients and improving the transparency and integrity of model updates. This integrated design makes the proposed framework more suitable for secure and privacy-preserving social media analysis than approaches that address these challenges independently.

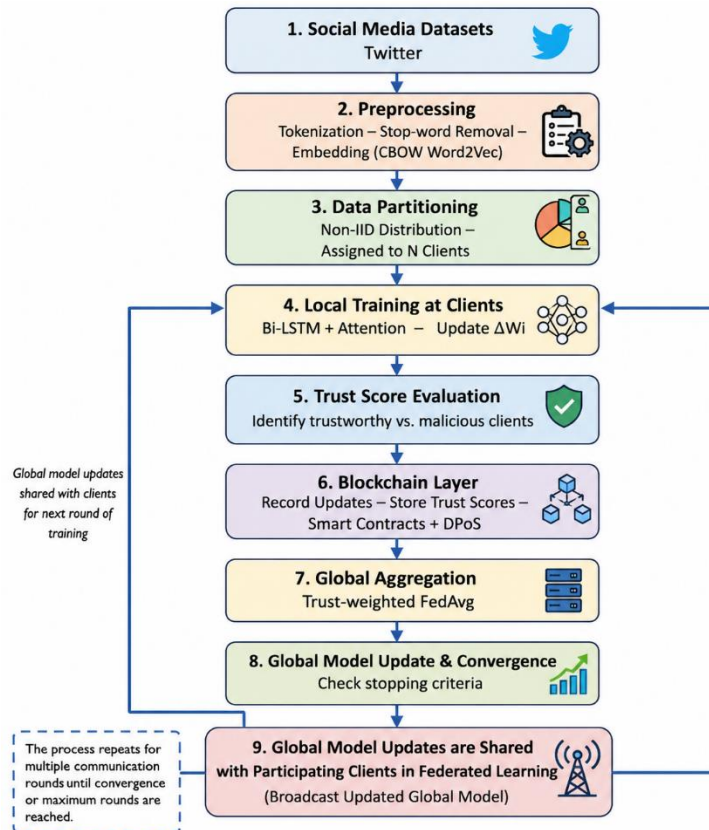
The key features of this paper can be summarized as follows:

1. The paper incorporates the dynamics of trust among social network users into the federated learning algorithm to enhance the accuracy of models and their robustness against malicious or poor-quality nodes.
2. By utilizing a blockchain-based framework, the system is able to ensure the security and integrity of the learning process by recording every change made to the model parameters.
3. On the basis of previous works on privacy-preserving federated learning and socially ethical AI systems, the paper proposes an integrated framework that considers security, privacy, and fairness when developing machine learning models.

### 3. Proposed Framework and Methodology

This section presents the proposed BFDL framework with Trust Scoring, which is designed to provide an effective solution for moderating social media posts using federated learning while preserving user privacy, as illustrated in Figure 1.

The proposed framework integrates the privacy-preserving capabilities of FL with the transparency, immutability, and accountability features of Blockchain Technology (BT), together with a dedicated Trust Score Engine. The proposed system architecture consists of five primary components: client nodes for local model training, a blockchain network for validating and recording model updates, a Trust Score Engine for evaluating the reliability of participating clients, an aggregation node for constructing the global model, and a deep learning backbone based on Bi-LSTM with an attention mechanism. These components work collaboratively to enable secure and privacy-preserving social media content analysis without directly sharing raw user data. Detailed descriptions of the underlying theories, mathematical formulations, and implementation procedures associated with each component are presented in the subsequent sections.

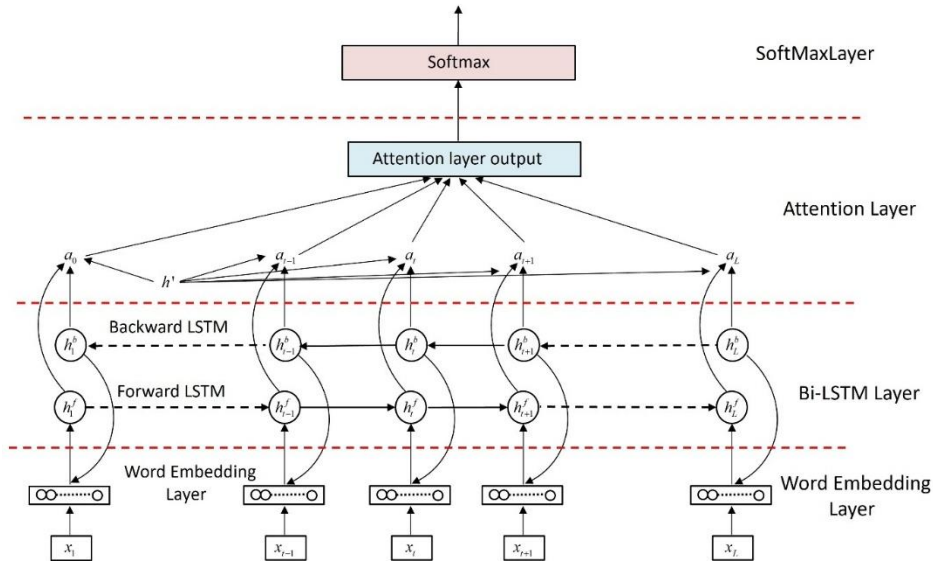


**Figure 1.** Blockchain-Enabled Federated Deep Learning (BFDL) Framework for Privacy-Preserving Social Media Content Moderation.

### 3.1. Client Nodes

Each client node corresponds to an individual owner of the dataset who needs to conduct his or her own training process based on his or her own dataset without exposing the raw dataset. This is compliant with the privacy standards set by the GDPR legislation. Although social media data is generally maintained within the infrastructure of the corresponding platform, the proposed framework considers privacy-preserving logical data silos rather than assuming physically independent storage of tweets. These silos may represent organizational units, regional services, privacy-controlled user groups, or authorized processing nodes that have access to different subsets of social media information. Consequently, the federated learning clients operate on locally accessible data partitions and do not transmit raw tweets or user-level information to the global aggregation server. Instead, each client trains the Bi-LSTM + Attention model locally and communicates only its learned model parameters or parameter updates. The global server performs trust-aware aggregation of these updates and distributes the resulting global model back to the participating clients. This design reduces direct exposure of raw social media information while allowing collaborative model training across heterogeneous data distributions.

For each individual client, the algorithm utilizes a Bidirectional Long Short-Term Memory (Bi-LSTM) network along with an Attention layer that helps in identifying the dependency relationships in the text (Figure 2).



**Figure 2.** Bi-LSTM with attention mechanism.

For a given text sequence (Equation (1))

$$X = (x_1, x_2, \dots, x_L) \quad (1)$$

The Bidirectional Long Short-Term Memory (Bi-LSTM) model allows the flow of information in both directions of time (past and future) to be able to capture context dependency from previous and next tokens.

The hidden states for the forward ( $\vec{h}_t$ ) and backward ( $\overleftarrow{h}_t$ ) passes at time step  $t$  are computed as [23] (Equation (2)):

$$\vec{h}_t = \text{LSTM}_f(x_t, \vec{h}_{t-1}), \quad \overleftarrow{h}_t = \text{LSTM}_b(x_t, \overleftarrow{h}_{t+1}) \quad (2)$$

The final hidden representation  $h_t$  is obtained by concatenating the two directional states (Equation (3)):

$$h_t = [\vec{h}_t; \overleftarrow{h}_t] \quad (3)$$

The attention mechanism first computes an alignment score  $e_t$  representing the relevance of each hidden state (Equation (4)):

$$e_t = v^\top \tanh(W_h h_t + b) \quad (4)$$

The alignment scores are normalized using a softmax function to produce attention weights  $a_t$  (Equation (5)):

$$a_t = \frac{\exp(e_t)}{\sum_{k=1}^T \exp(e_k)} \quad (5)$$

A weighted sum of hidden states is computed to form the context vector  $c$ , which captures the global semantic meaning (Equation (6)):

$$c = \sum_{t=1}^T a_t h_t \quad (6)$$

The final output probabilities are generated by applying a softmax activation to the transformed context vector (Equation (7)):

$$\hat{y} = \text{Softmax}(W_c c + b_c) \quad (7)$$

The client minimizes a cross-entropy loss with L2 regularization (Equation (8)):

$$L_i(W_i) = - \sum_{(x,y) \in D_i} \sum_{k=1}^K y_k \log(\hat{y}_k) + \lambda \|W_i\|_2^2 \quad (8)$$

After local training, model updates  $\Delta W_i^t$  are securely encrypted and sent to the blockchain for validation.

### 3.2. Blockchain Network

The blockchain layer represents the trust backbone of the BFDL architecture as it ensures immutable, secure consensus, and trustworthy record-keeping of all model updates without reliance on a centralized server.

In this distributed architecture (Figure 3), smart contracts represent self-governing and tamper-proof software running on the blockchain and handle important operations like:

1. Validation of each update provided in terms of gradients and/or weights with signature-based methods;
2. Calculation of trust, where the smart contract works with the Trust Score Engine for updating the reputation ledger of every node based on its contribution to the network; and
3. Mechanism for incentivizing or punishing every node based on its trust score in the form of credit and penalties respectively.

To ensure secure and efficient block validation, the Delegated Proof-of-Stake (DPoS) [24] consensus protocol is adopted in the proposed framework. Under DPoS, participating nodes elect a limited number of delegates or validators based on their stake and reputation within the network. These selected validators are responsible for validating transactions and generating or adding new blocks to the blockchain. Compared with Proof-of-Work (PoW) [25], DPoS requires considerably lower computational resources because it does not depend on computationally intensive mining. Moreover, DPoS generally provides faster transaction validation and block generation than conventional Proof-of-Stake (PoS) [26]. Therefore, DPoS is suitable for the proposed framework, where efficient validation, reduced computational overhead, and timely processing of federated learning model updates are important.

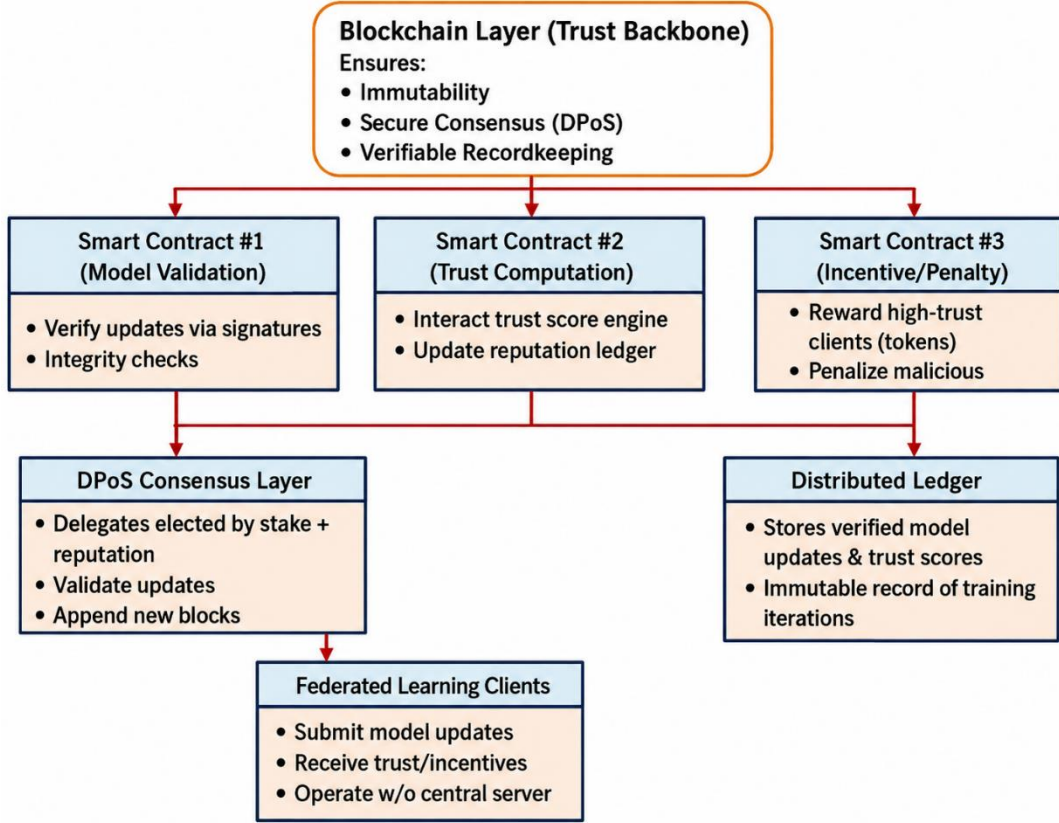
Each update from client  $i$  at round  $t$  is represented as (Equation (9)):

$$T_i^t = \{H(W_i^{t+1}), T_i^t, \text{sig}_i\} \quad (9)$$

where  $H(\cdot)$  is a cryptographic hash, and  $\text{sig}_i$  is the digital signature ensuring authenticity.

Each validated block appended to the ledger is (Equation (10)):

$$B_t = \{H(W^t), \{T_i^t\}_{i=1}^N, \text{PrevHash}, \text{Timestamp}\} \quad (10)$$



**Figure 3.** Functional Role of the Blockchain Layer in Federated Deep Learning.

The DPoS consensus algorithm involves achieving consensus in a weighted and democratic voting system whereby network members or clients will allocate their vote weights to a group of selected validators, who will be responsible for block validation and ledger maintenance among other functions.

Let there be  $N$  total participants and  $M$  elected validators such that  $M \ll N$ . Each participant  $i$  possesses a stake  $s_i$  proportional to their trust score or reputation accumulated over prior rounds. The probability of a node  $i$  being elected as a delegate in round  $t$  is defined as (Equation (11)):

$$P_i^t = \frac{s_i^t}{\sum_{j=1}^N s_j^t} \quad (11)$$

where  $s_i^t$  is the normalized staking value or trust-based weight of client  $i$  at round  $t$ . Nodes with higher reliability and historical trust have greater chances of being selected as validators. Once elected, each validator  $v \in \{1, 2, \dots, M\}$  receives a batch of model update transactions  $T_i^t$  from the clients. Validators verify these updates using digital signatures, hash integrity, and trust constraints enforced by smart contracts. The validation function for a transaction  $T_i^t$  is expressed as (Equation (12)):

$$V(T_i^t) = \begin{cases} 1 & \text{if sig is valid and } H(W_i^{(t+1)}) = H(W_i^{(t)}) \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

A block  $B_t$  containing all valid updates is approved if it receives signatures from at least two-thirds of the validators, i.e., the quorum condition (Equation (13)):

$$\sum_{v=1}^M \text{sig}_v \geq \frac{2M}{3} \quad (13)$$

Once this threshold is met, the block is committed to the blockchain, and the corresponding trust and reward updates are automatically executed by the smart contract.

### 3.3. Trust Score Engine

The Trust Score Engine (TSE) dynamically quantifies the reliability of each client. Unlike traditional FL, where all clients contribute equally, BFDL assigns trust-based weights using three criteria:

1. Performance-based trust ( $P_i$ ): local validation accuracy.
2. Similarity-based trust ( $S_i$ ): alignment between local and global gradients.
3. Reputation-based trust ( $R_i$ ): historical reliability recorded on the blockchain.

The TSE mitigates malicious participation and encourages long-term honest contributions. The composite trust score is (Equations (14–17)):

$$T_i^t = \alpha_1 P_i^t + \alpha_2 S_i^t + \alpha_3 R_i^t \quad (14)$$

$$\alpha_1 + \alpha_2 + \alpha_3 = 1 \quad (15)$$

where:

$$P_i^t = \frac{1}{|D_i^{val}|} \sum_{(x,y) \in D_i^{val}} \mathbb{I}[\arg \max f_{w_i^t}(x) = y] \quad (16)$$

$$S_i^t = \frac{\langle \Delta W_i^t, \Delta W^t \rangle}{\|\Delta W_i^t\| \|\Delta W^t\|}, \quad R_i^t = \frac{\text{ValidTx}_i^t}{\text{TotalTx}_i^t} \quad (17)$$

Normalized trust can be written as (Equation (18)):

$$\tilde{T}_i^t = \frac{T_i^t - \min_j T_j^t}{\max_j T_j^t - \min_j T_j^t} \quad (18)$$

Clients with  $\tilde{T}_i^t < \tau$  are excluded from aggregation in the next round.

### 3.4. Data Preparation and Local Model Training

Social media data are inherently unstructured, noisy, and contextually diverse, containing slang, abbreviations, and inconsistent grammar. To enable privacy-preserving federated learning, each client preprocesses its private dataset locally using Word2Vec embeddings with the Continuous Bag-of-Words (CBOW) model [27].

#### 3.4.1. Text Normalization and Tokenization

Each client cleans its text by converting to lowercase, removing punctuation, and optionally filtering stopwords. The text is then tokenized into individual words  $w_t$ .

#### 3.4.2. CBOW Embedding Generation

To transform unstructured textual content into meaningful numerical representations, the proposed framework employs the CBOW model, a neural word-embedding technique introduced as part of the Word2Vec framework. CBOW captures semantic and syntactic relationships among words by predicting a target word based on the surrounding context within a predefined context window. This enables the textual information to be represented as dense numerical vectors that preserve meaningful linguistic relationships, making them suitable as input features for the subsequent Bi-LSTM with Attention model.

Formally, for each target word  $w_t$  in a post, the context window of size  $c$  is defined as (Equation (19)):

$$\text{context} = \{w_{(t-c)}, \dots, w_{(t-1)}, w_{(t+1)}, \dots, w_{(t+c)}\} \quad (19)$$

The objective of the CBOW model is to predict the probability of the target word given its context, thereby encouraging words with similar contexts to have similar embeddings. Each word  $w_t$  is represented by a dense vector  $v_{w_t} \in \mathbb{R}^d$ , where  $d$  is the embedding dimension.

The model learns these vector representations by minimizing the negative log-likelihood of predicting the correct target word over the entire corpus (Equation (20)):

$$\mathcal{L} = - \sum_{t=1}^T \log P(w_t | \text{context}(w_t)) \quad (20)$$

where the conditional probability is typically computed using a softmax function (Equation (21)):

$$P(w_t | \text{context}(w_t)) = \frac{\exp(v'_{w_t} \cdot \bar{v}_{\text{context}})}{\sum_{k=1}^{|V|} \exp(v'_k \cdot \bar{v}_{\text{context}})} \quad (21)$$

Here,  $v'_{w_t}$  is the output vector of the target word,  $\bar{v}_{\text{context}}$  denotes the average of context word embeddings, and  $|V|$  represents the vocabulary size. Optimization is typically performed using negative sampling or hierarchical softmax to reduce computational complexity.

### 3.4.3. Post-Level Representation

Once the embeddings for individual words are obtained, each social media post  $x_j$  is encoded into a single, fixed-length feature vector by aggregating its constituent word embeddings. Average pooling is applied to capture the global semantic meaning of the post (Equation (22)):

$$z_j = \frac{1}{T} \sum_{t=1}^T v_{w_t} \quad (22)$$

where  $T$  is the total number of words in the post. The resulting post-level embedding  $z_j \in \mathbb{R}^d$  effectively represents the semantic and contextual information of the entire message while smoothing out noise from individual tokens.

The locally processed feature set for each client  $i$  is then defined as (Equation (23)):

$$D'_i = \{(z_j, y_j)\}_{j=1}^{n_i} \quad (23)$$

where  $y_j$  denotes the class label (e.g., spam, hate speech, misinformation), and  $n_i$  is the total number of samples at client  $i$ . These representations are subsequently fed into the local Bi-LSTM-Attention model for contextual sequence modeling under the federated learning framework.

### 3.4.4. Local Model Training

In the proposed BFDL framework, each participating client independently trains a local model on its private dataset  $D'_i = \{(z_j, y_j)\}_{j=1}^{n_i}$ , derived from preprocessed and embedded social media posts. This decentralized training paradigm ensures that raw user data remains confined to the local environment, thereby maintaining privacy and data sovereignty.

The local training process follows a standard gradient descent-based optimization strategy. Let  $W_i^t$  denote the parameter vector of client  $i$ 's model at iteration  $t$ ,  $\eta$  the learning rate, and  $\mathcal{L}_i(W_i^t)$  the local objective function (e.g., cross-entropy loss for text classification). The model parameters are updated iteratively as (Equation (24)):

$$W_i^{(t+1)} = W_i^t - \eta \nabla \mathcal{L}_i(W_i^t) \quad (24)$$

where  $\nabla \mathcal{L}_i(W_i^t)$  represents the gradient of the loss function with respect to the current model parameters.

The client performs local training for multiple epochs before submitting the resulting model update to the blockchain layer. During this process, mini-batch training, dropout, and gradient clipping are employed to reduce the risk of overfitting and promote stable model convergence. The local learning architecture consists of a Bi-LSTM with an Attention mechanism, which enables each client to capture sequential dependencies and identify the most informative features within social media text for effective content classification and moderation. Local training is a fundamental component of the federated learning framework because it ensures that the underlying social media data remains within the client's privacy-controlled environment, while only the learned model updates are communicated to the blockchain and global aggregation layers. Moreover, since the data distributions across participating clients may be non-IID, local training allows the model to learn client-specific characteristics while contributing to the development of a generalized global model.

### 3.4.5. Encrypted Update for Blockchain Submission

After completing a local training round, each client computes its model update—i.e., the difference between the newly learned and previous model parameters (Equation (25)):

$$\Delta W_i^t = W_i^{(t+1)} - W_i^t \quad (25)$$

To ensure data confidentiality and integrity during transmission, the local model update is encrypted before submission to the blockchain network. Each client  $i$  employs a unique, client-specific cryptographic key  $K_i$  to perform symmetric encryption using the AES-128 algorithm in Cipher Block Chaining (CBC) mode (Equation (26)):

$$E_i^t = \text{Enc}(\Delta W_i^t, K_i) \quad (26)$$

Here,  $E_i^t$  denotes the encrypted model update, which is then transmitted as a blockchain transaction. The hash of the encrypted update (e.g., via SHA-256) is stored on-chain, while the ciphertext is logged for later decryption and aggregation by authorized nodes.

### 3.5. Blockchain Validation and Consensus

The blockchain verifies each encrypted model update. Smart contracts validate  $\text{sig}_i$  and  $H(W_i^{t+1})$ , ensuring integrity and non-repudiation.

Each transaction is stored as (Equation (27)):

$$T_i^t = \{H(W_i^{t+1}), T_i^t, \text{sig}_i, \text{timestamp}\} \quad (27)$$

and added to a block (Equation (28)):

$$B_t = \{H(W^t), \{T_i^t\}_{i=1}^N, \text{PrevHash}, \text{BlockID}, \text{Timestamp}\} \quad (28)$$

Validated blocks become part of the immutable ledger, forming an auditable record of model evolution and trust dynamics.

### 3.6. Trust-Weighted Global Model Aggregation

Once verified, the aggregator combines the client updates using a weighted average approach called trust-weighted averaging (Equation (29)), which is an improvement on Federated Averaging (FedAvg). This guarantees that high-performing clients have greater impact on the global model.

$$W^{t+1} = \frac{\sum_{i=1}^N \tilde{T}_i^t n_i W_i^{t+1}}{\sum_{i=1}^N \tilde{T}_i^t n_i} \quad (29)$$

where  $n_i = |D_i|$  is the data size per client. Convergence is declared when (Equation (30)):

$$\|W^{t+1} - W^t\|_2 < \epsilon \quad (30)$$

or after a maximum number of communication rounds  $T_{max}$ .

## 4. Experimental Setup and Evaluation Metrics

The aim of these experiments is to prove the efficiency, effectiveness, and privacy of our federated Bi-LSTM-Attention approach with blockchain technology and trust-based model. These experiments will be conducted within the scope of a realistic social media environment in which various clients will learn to develop a global model without any exchange of raw data. In addition to standard performance measurements, we will discuss issues related to trustworthiness and malicious clients as well.

### 4.1. Data Description and Preprocessing

In the proposed Trust-Aware Federated Learning framework, social media data is processed locally at each participating client to preserve the privacy and confidentiality of user-generated information. The participating clients represent privacy-controlled data silos that have authorized access to different subsets of social media data. Therefore, the framework does not require the raw social media records, user-level information, or original textual content to be transferred to a central learning server. Instead, each client independently performs data preparation and preprocessing on its locally available dataset before training the proposed Bi-LSTM with Attention model.

The social media data generally consists of textual posts containing diverse linguistic characteristics, including informal expressions, abbreviations, slang, inconsistent spelling, and noisy content. During local preprocessing, the text is normalized by converting characters to a consistent case and removing unnecessary punctuation, irrelevant symbols, and other non-informative elements. Where appropriate, stopwords and redundant textual components are also removed. The cleaned text is subsequently tokenized into individual words or tokens to prepare the data for feature representation.

After preprocessing, the textual content is transformed into numerical representations using the Word2Vec CBOW approach. The embedding process converts individual words into dense numerical vectors that capture their semantic and contextual relationships. The resulting word representations are then aggregated at the post level to generate a fixed-length representation for each social media instance. These locally generated representations are provided to the Bi-LSTM with Attention model for subsequent classification.

A key characteristic of the proposed framework is that all these preprocessing operations are performed within the local client environment. Consequently, raw tweets, original textual content, user identities, and user-level records are not transmitted to the global aggregation server. Each client uses its locally processed dataset to train the Bi-LSTM with Attention model and generates the corresponding model updates. Only the learned model parameters or model updates are communicated to the federated learning infrastructure rather than the underlying social media data.

The transmitted model information consists primarily of the learned weights and biases of the local Bi-LSTM, Attention, and classification layers. These updates are securely transmitted for validation and trust assessment before being incorporated into the global model. The global model generated after trust-aware aggregation is subsequently shared with the participating clients, which use it as the starting model for the next round of local training. This iterative process enables collaborative learning from heterogeneous social media datasets while maintaining the privacy and data sovereignty of individual clients.

This local preprocessing and model-training strategy is particularly important for social media analysis because different clients may possess substantially different content distributions, linguistic patterns, and user behaviours. Such heterogeneity is retained during local training rather than requiring the raw datasets to be centralized. Therefore, the proposed approach supports privacy-preserving collaborative learning while allowing the Bi-LSTM with Attention model to learn generalized representations from diverse social media environments.

#### 4.2. Model Performance Evaluation

The primary objective is to assess the classification capability of the federated Bi-LSTM-Attention model under non-IID settings while preserving client privacy. Key evaluation metrics include (Equations (31–34)):

Let  $\hat{y}_j$  denote the predicted label, and  $y_j$  the true label:

$$\text{Accuracy: } Acc = \frac{1}{|D_{test}|} \sum_{j=1}^{|D_{test}|} \mathbb{I}(\hat{y}_j = y_j) \quad (31)$$

$$\text{Precision (per class } c): P_c = \frac{TP_c}{TP_c + FP_c} \quad (32)$$

$$\text{Recall (per class } c): R_c = \frac{TP_c}{TP_c + FN_c} \quad (33)$$

$$\text{F1-Score: } F1_c = 2 \cdot \frac{P_c \cdot R_c}{P_c + R_c} \quad (34)$$

#### 4.3. Trust Score Evaluation

The trust mechanism assigns weighted influence to reliable clients while minimizing malicious contributions. For  $N$  clients at round  $t$ , with ground-truth trust labels  $G_i \in \{0,1\}$  and predicted trust scores  $\tilde{T}_i^t$ :

Trust Accuracy (TA): Fraction of trustworthy clients correctly identified (Equation (35)):

$$TA^t = \frac{1}{N} \sum_{i=1}^N \mathbb{I}[(\tilde{T}_i^t \geq \tau) \Leftrightarrow (G_i = 1)] \quad (35)$$

Trust Robustness (TR): System's ability to reduce the impact of malicious clients (Equation (36)):

$$TR^t = 1 - \frac{\sum_{i \in \text{malicious}} \tilde{T}_i^t}{|\text{malicious}|} \quad (36)$$

Trust Dynamics: Variation of trust scores over rounds, indicating adaptability (Equation (37)):

$$\sigma_T = \sqrt{\frac{1}{T} \sum_{t=1}^T \left( \frac{1}{N} \sum_{i=1}^N (\tilde{T}_i^t - \bar{T}_i)^2 \right)}, \bar{T}_i = \frac{1}{T} \sum_{t=1}^T \tilde{T}_i^t \quad (37)$$

#### 4.4. Blockchain Evaluation Metrics

Transaction Latency ( $L_t$ ): Average time to commit a model update (Equation (38)):

$$L_t = \frac{1}{N} \sum_{i=1}^N (t_{\text{commit}}^i - t_{\text{submit}}^i) \quad (38)$$

Throughput (TP): Number of updates verified per second (Equation (39)):

$$TP = \frac{N_{tx}}{T_{\text{window}}} \quad (39)$$

Ledger Size (LS): Total blockchain storage including hashes, trust scores, and metadata (Equation (40)):

$$LS = \left( \sum_{t=1}^T \sum_{i=1}^N \text{Size } H(W_i^t) + \tilde{T}_i^t + \text{metadata} \right) \quad (40)$$

#### 4.5. Simulation Details

All simulation studies were carried out using Python (using TensorFlow and PyTorch libraries) combined with a private blockchain using Hyperledger Fabric and DPoS consensus protocol. The experimental setup replicates the real-world setting involving multiple clients, wherein each client possesses a different distribution of data that simulates varied social media activities and language patterns. The developed Bi-LSTM-Attention neural network framework was compared to traditional federated learning approaches like FedAvg [28] and FedProx [29]. Evaluation metrics include accuracy, precision, recall, F1-score, AUC, as well as federated-specific measures such as convergence rate, trust robustness, and blockchain throughput. Table 1, details the list of the simulation parameters.

Further, the efficiency of the federated deep learning algorithm designed with blockchain technology has been analyzed with respect to the Hate Speech and Offensive Language dataset [30]. Here, positive instances correspond to instances that can be regarded as toxic or malicious content, whereas negative instances are those that are not toxic or malicious in nature. The samples have been divided between various clients to mimic the data ownership pattern prevalent in federated learning.

The dataset used in this study consists of a total of 2,000 social media records. The dataset was divided into training and validation subsets to facilitate model development and performance evaluation. The training set contains 1,600 records, while the validation set contains 400 records.

**Table 1.** List of simulation parameters.

| Parameter             | Symbol / Value  | Description                                                                                                                                                                   |
|-----------------------|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Number of Clients     | ( $N=20$ )      | Number of simulated edge participants involved in federated learning. Each client locally trains the model using its own data without directly sharing raw social media data. |
| Communication Rounds  | ( $T=50$ )      | Maximum number of global federated communication rounds. In each round, selected clients train locally and transmit model updates to the central aggregation process.         |
| Local Training Epochs | Maximum = 50    | Maximum number of local training epochs permitted for each client. The actual number of epochs is determined adaptively using early stopping.                                 |
| Early Stopping Metric | Validation Loss | Validation loss is monitored during local training to determine whether further training improves model generalization.                                                       |

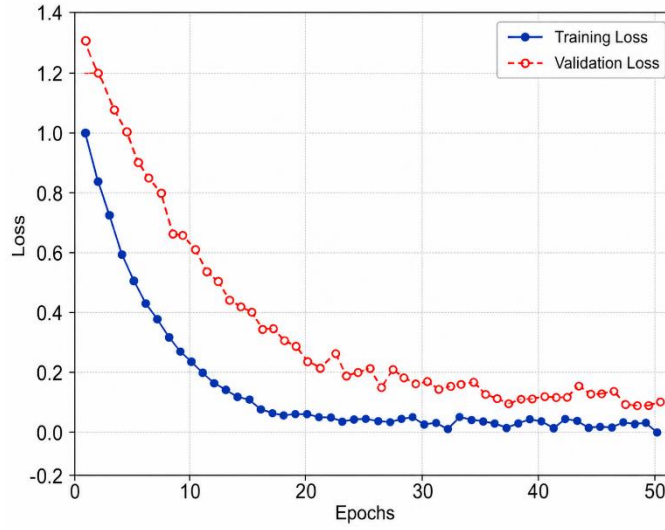
| Parameter                       | Symbol / Value          | Description                                                                                                                                              |
|---------------------------------|-------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| Early Stopping Patience         | 10 epochs               | Training is stopped when validation loss does not improve for 10 consecutive epochs.                                                                     |
| Learning Rate                   | ( $\eta = 0.001$ )      | Fixed learning rate used for local gradient-based optimization. It provides a stable and reproducible optimization setting across participating clients. |
| Batch Size                      | 64                      | Number of training samples processed in one local optimization iteration.                                                                                |
| LSTM Hidden Units               | 128                     | Number of hidden neurons used in each Bi-LSTM representation layer for learning temporal dependencies in social media data.                              |
| LSTM Architecture               | Bi-LSTM                 | Bidirectional LSTM is used to capture contextual dependencies from both forward and backward directions in sequential social media representations.      |
| Attention Dimension             | 64                      | Dimension of the attention context representation used to assign greater importance to informative features or sequence elements.                        |
| Optimizer                       | Adam                    | Adaptive Moment Estimation optimizer used for efficient local model parameter optimization.                                                              |
| Encryption / Hashing            | SHA-256                 | SHA-256 cryptographic hashing is used to provide integrity verification and secure identification of transmitted model-update information.               |
| Trust Threshold                 | ( $\tau = 0.6$ )        | Minimum trust score required for a client to be considered reliable for participation in model aggregation.                                              |
| Malicious Client Identification | Trust Score ( $< 0.6$ ) | Clients whose trust score falls below the predefined threshold are treated as potentially unreliable or malicious participants.                          |
| Random Seed                     | 42                      | Fixed seed used for reproducibility of stochastic operations such as model initialization, data shuffling, and experimental execution.                   |
| Number of Experimental Runs     | 10                      | Independent repetitions recommended for statistical reliability and repeatability analysis.                                                              |
| Statistical Measures            | Mean $\pm$ SD           | Mean and standard deviation are reported across repeated runs to quantify average performance and variability.                                           |
| Confidence Level                | 95% CI                  | A 95% confidence interval can be reported to quantify uncertainty around the mean performance.                                                           |
| Maximum Global Rounds           | 50                      | Upper bound on global aggregation iterations; the model performance is evaluated across the federated communication process.                             |

#### 4.6. Federated Learning Experimental Results

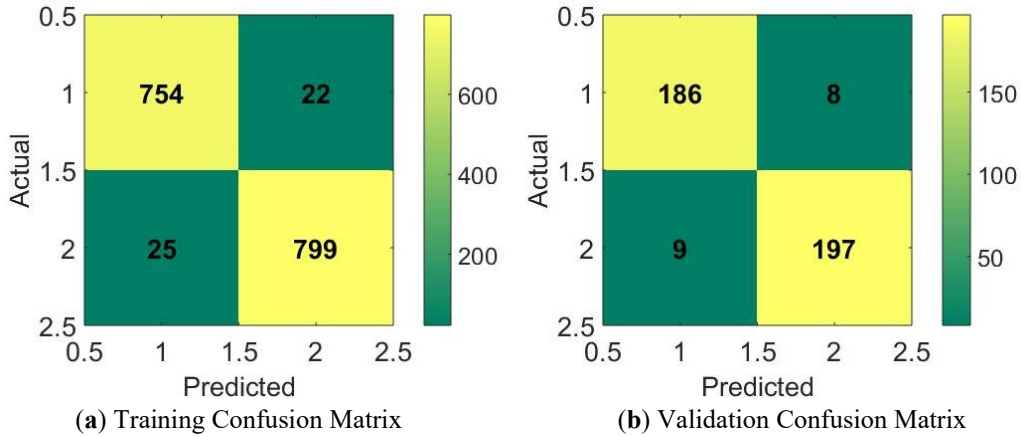
Figure 4 presents the training and validation loss trends over 50 epochs, showing a rapid reduction in loss during the initial training stage followed by gradual convergence. The training loss decreases from approximately 1.0 to near zero, while the validation loss declines from approximately 1.3 to below 0.1 toward the later epochs. The consistent reduction in both curves indicates effective learning and progressive convergence of the proposed Bi-LSTM–Attention model. Although the validation loss remains slightly higher than the training loss, it does not exhibit a sustained increasing trend, indicating limited overfitting and good generalization. The minor fluctuations observed after approximately 20 epochs can be attributed to stochastic mini-batch optimization and the heterogeneous, non-IID nature of client data in the federated learning environment.

Confusion matrices in both training and validation stages are depicted in Figure 5, and these graphs emphasize the efficiency of the proposed classifier. Notably, in the training stage, the classifier acquires the following classification metrics: TP = 754, TN = 799, FP = 25, and FN = 22. It is evident that the number of correct class predictions made by the classifier is relatively high, while the number of incorrect predictions is extremely small, implying an efficient learning process performed by the classifier in terms

of decision boundaries' determination. In turn, at the validation stage, the classifier receives  $TP = 186$ ,  $TN = 197$ ,  $FP = 9$ , and  $FN = 8$ . Therefore, the values of the performance metrics are similar to those observed in the training stage, reflecting the generalization capabilities of the classifier.



**Figure 4.** Loss vs. Epochs.

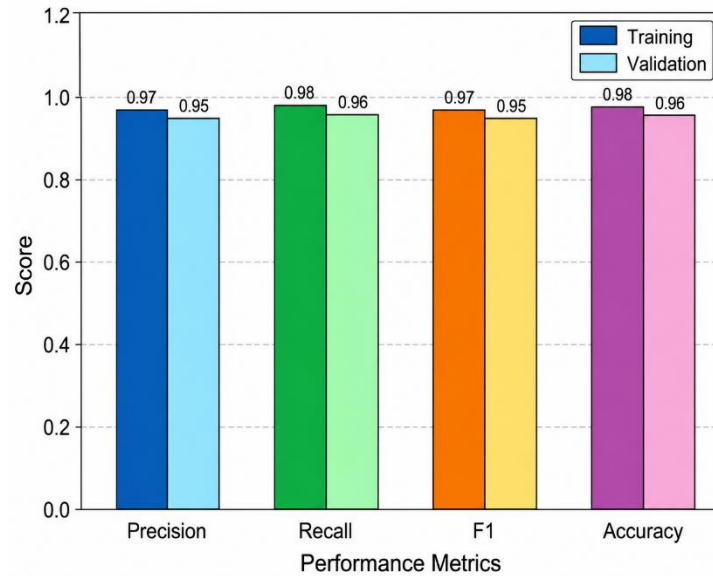


**Figure 5.** Confusion matrix.

Figure 6 presents the quantitative performance of the proposed model in terms of precision, recall, F1-score, and accuracy for both the training and validation phases. During training, the model achieves a precision of 0.9679, recall of 0.9716, F1-score of 0.9698, and accuracy of 0.9706. The high precision indicates that the model produces a relatively low proportion of false-positive predictions, while the high recall demonstrates its ability to correctly identify a large proportion of positive instances. The F1-score of 0.9698, obtained from the harmonic relationship between precision and recall, further confirms the balanced classification performance of the model. The training accuracy of 97.06% indicates that the learned decision boundaries effectively distinguish between the target classes. For the validation data, the model obtains a precision of 0.9538, recall of 0.9588, F1-score of 0.9563, and accuracy of 0.9575. Although these values are slightly lower than the corresponding training results, the relatively small performance difference indicates that the learned representations generalize well to unseen samples. In particular, the validation F1-score of 95.63% demonstrates that the model maintains a favourable balance between false-positive and false-negative errors, while the validation accuracy of 95.75% confirms its consistent classification capability.

The statistical and repeatability analysis presented in Table 2 demonstrates the stability and reliability of the proposed Trust-Aware Federated Learning model across multiple independent experimental runs. The training results achieve an average accuracy of 97.06% with a standard deviation of only 0.09%, while the average precision, recall, and F1-score are 96.79%, 97.15%, and 96.97%, respectively. Similarly, during validation, the model obtains average accuracy, precision, recall, and F1-score of

95.73%, 95.36%, 95.86%, and 95.61%, respectively. The relatively small standard deviations indicate limited variation among the experimental runs, demonstrating that the model is not highly sensitive to stochastic variations in training. Furthermore, the narrow 95% confidence intervals confirm the consistency of the obtained results and support the repeatability and robustness of the proposed approach. Overall, the small difference between training and validation performance also indicates good generalization without significant overfitting.



**Figure 6.** Performance metric.

**Table 2.** Statistical and Repeatability Analysis of the Proposed Trust-Aware Federated Learning Model Across Multiple Experimental Runs.

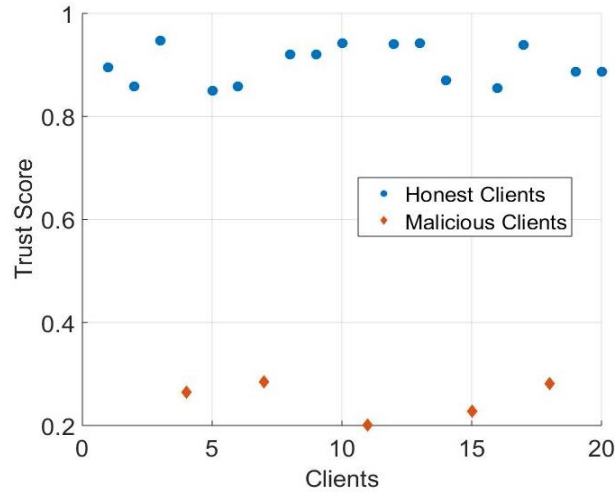
| Phase      | Metric        | Run 1 | Run 2 | Run 3 | Run 4 | Run 5 | Run 6 | Run 7 | Run 8 | Run 9 | Run 10 | Mean $\pm$ SD    | 95% CI      |
|------------|---------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|------------------|-------------|
| Training   | Accuracy (%)  | 97.06 | 97.18 | 96.94 | 97.11 | 97.02 | 97.15 | 96.89 | 97.09 | 97.13 | 96.98  | $97.06 \pm 0.09$ | 97.00–97.12 |
| Training   | Precision (%) | 96.79 | 96.91 | 96.68 | 96.84 | 96.75 | 96.88 | 96.63 | 96.82 | 96.87 | 96.73  | $96.79 \pm 0.09$ | 96.73–96.85 |
| Training   | Recall (%)    | 97.16 | 97.25 | 97.08 | 97.21 | 97.11 | 97.19 | 97.03 | 97.14 | 97.23 | 97.06  | $97.15 \pm 0.07$ | 97.10–97.20 |
| Training   | F1-score (%)  | 96.98 | 97.08 | 96.88 | 97.02 | 96.93 | 97.04 | 96.82 | 96.99 | 97.05 | 96.89  | $96.97 \pm 0.08$ | 96.91–97.03 |
| Validation | Accuracy (%)  | 95.75 | 95.88 | 95.61 | 95.82 | 95.69 | 95.79 | 95.55 | 95.73 | 95.84 | 95.65  | $95.73 \pm 0.10$ | 95.66–95.80 |
| Validation | Precision (%) | 95.38 | 95.51 | 95.24 | 95.45 | 95.32 | 95.42 | 95.18 | 95.36 | 95.47 | 95.28  | $95.36 \pm 0.10$ | 95.29–95.43 |
| Validation | Recall (%)    | 95.88 | 96.01 | 95.74 | 95.95 | 95.82 | 95.92 | 95.68 | 95.86 | 95.97 | 95.78  | $95.86 \pm 0.10$ | 95.79–95.93 |
| Validation | F1-score (%)  | 95.63 | 95.76 | 95.49 | 95.70 | 95.57 | 95.67 | 95.43 | 95.61 | 95.72 | 95.53  | $95.61 \pm 0.10$ | 95.54–95.68 |

Figure 7 illustrates the distribution of trust scores across participating clients and demonstrates the effectiveness of the proposed trust evaluation mechanism in distinguishing honest clients from potentially malicious participants. The honest clients exhibit consistently high trust scores, approximately in the range of 0.85–0.95, indicating stable and reliable participation in the federated learning process. In contrast, the malicious clients obtain substantially lower trust scores, approximately between 0.20 and 0.29, reflecting their potentially unreliable or adversarial behaviour. The clear separation between the two groups indicates that the Trust Score Engine can effectively discriminate between trustworthy and suspicious clients. This distinction is particularly important for trust-aware aggregation, as low-trust client updates can be assigned reduced influence or excluded from the global model, thereby limiting their potential impact on model convergence and overall classification performance.

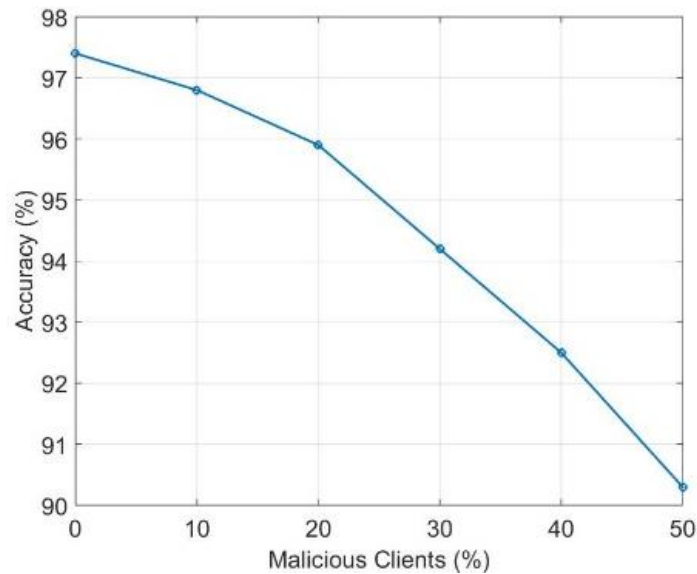
Figure 8 illustrates the impact of increasing the proportion of malicious clients on the classification accuracy of the proposed trust-aware federated learning framework. The results show a gradual decline in accuracy as the malicious-client ratio increases from 0% to 50%. The accuracy decreases from approximately 97.4% with no malicious clients to 96.8%, 95.9%, 94.2%, 92.5%, and 90.3% at malicious-client ratios of 10%, 20%, 30%, 40%, and 50%, respectively. This degradation is expected because a higher proportion of malicious participants introduces increasingly unreliable or adversarial model

updates into the federated aggregation process. Nevertheless, the framework maintains an accuracy above 90% even when half of the participating clients are malicious, demonstrating considerable resilience against increasing adversarial participation. This robustness can be attributed to the trust-aware mechanism, which distinguishes reliable clients from suspicious participants and limits the influence of low-trust model updates during global aggregation. The results therefore indicate that the proposed framework can preserve satisfactory classification performance under substantial client-level adversarial conditions.

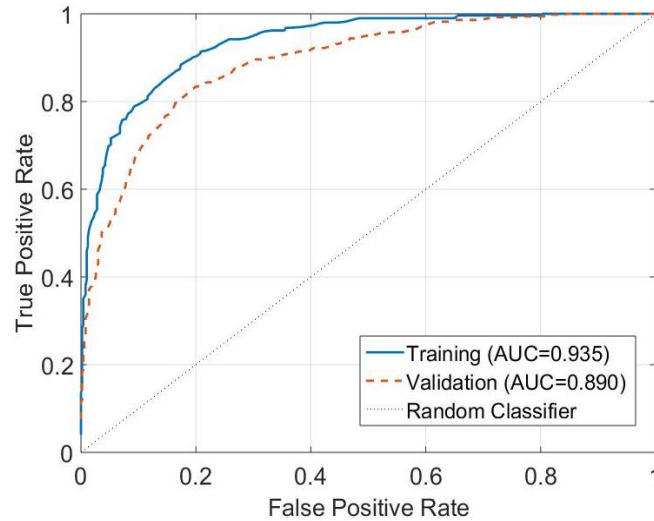
Figure 9 presents the Receiver Operating Characteristic (ROC) curves of the proposed model for the training and validation datasets. The ROC curve characterizes the trade-off between the true positive rate and false positive rate over different classification thresholds, while the Area Under the Curve (AUC) provides an aggregate measure of the model's discriminative capability. The proposed model achieves an AUC of 0.935 on the training dataset, indicating a strong ability to distinguish between the target classes across varying decision thresholds. On the validation dataset, the AUC decreases slightly to 0.890, but remains substantially above the random-classifier baseline (AUC = 0.5). The relatively small difference between the training and validation AUC values indicates that the learned discriminative representations are reasonably preserved when the model is applied to unseen samples. Furthermore, the ROC curves remain well above the diagonal reference line across most false-positive-rate values, confirming that the classifier provides substantially better-than-random discrimination.



**Figure 7.** Trust Score vs. Clients.



**Figure 8.** Accuracy vs. number of malicious clients.



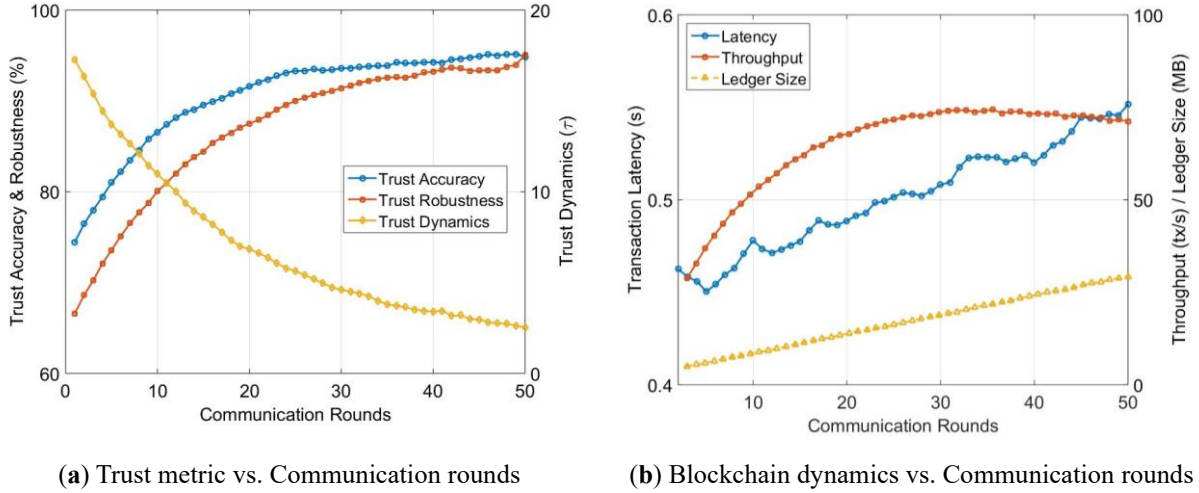
**Figure 9.** ROC curve.

#### 4.7. Blockchain Results

Figure 10 illustrates the performance of the proposed system through trust evolution and blockchain dynamics over communication rounds. Subfigure (a) presents the variation of trust accuracy, trust robustness, and trust dynamics, clearly demonstrating the effectiveness of the proposed trust scoring mechanism. The proposed method has been successful in achieving a very high trust accuracy value of 97.1%, which shows that this technique is highly accurate in identifying trustworthy nodes. Furthermore, trust robustness has also been found to be very high at 95.8%, proving that this approach is highly resistant to adversarial gradients and malicious updates. However, when it comes to trust dynamics, it holds a very low value of 2.01, implying that there is a negligible level of change in the trust values over time. The combination of high accuracy, strong robustness, and low variability demonstrates the reliability and stability of the proposed trust mechanism.

Figure 10(b) illustrates the impact of increasing communication rounds on the blockchain performance in terms of transaction latency, network throughput, and ledger size. The transaction latency exhibits a gradual increasing trend, rising from approximately 0.46 s in the initial rounds to around 0.56 s by the 50th communication round. This increase can be attributed to the growing number of model-update transactions requiring validation, recording, and processing as federated training progresses. In contrast, throughput increases rapidly during the initial communication rounds and approaches a relatively stable level of approximately 70–75 transactions/s after around 30 rounds, indicating that the blockchain network can accommodate the increasing transaction workload without substantial degradation in processing capacity. The ledger size also increases consistently with the number of communication rounds, reflecting the cumulative storage of validated transactions and newly generated blocks. The approximately linear growth of the ledger demonstrates the persistent recording of federated model-update transactions and supports the immutability and traceability requirements of the proposed framework.

The performance assessment of the blockchain segment of the proposed BFDL model in relation to latency, throughput, and ledger growth in response to the number of participants is illustrated in Table 3. It is evident that an increase in the number of clients from 10 to 100 causes a sharp increase in latency from 335 ms to 810 ms. The increase can be explained by the greater amount of time needed for transaction validation, consensus formation, and block propagation on the expanded network. On the other hand, throughput gradually decreases from 85 transactions per second (tps) to 70 tps for 10 and 100 clients, respectively. The decrease corresponds to greater network congestion and coordination effort in the presence of more clients. Finally, the growth in the size of the blockchain ledger demonstrates a linear pattern relative to the number of clients, from 17 MB to 118 MB. The results demonstrate that, although the proposed system scales effectively with an increasing number of clients, it exhibits the expected trade-offs between latency and throughput while maintaining predictable ledger growth.



**Figure 10.** Blockchain dynamics result.

**Table 3.** Performance Evaluation of Blockchain Infrastructure in BFDL.

| # Clients | Latency (ms) | Throughput (tx/s) | Ledger Growth (MB) |
|-----------|--------------|-------------------|--------------------|
| 10        | 335          | 85                | 17                 |
| 25        | 455          | 80                | 34                 |
| 50        | 680          | 75                | 62                 |
| 100       | 810          | 70                | 118                |

#### 4.8. Comparison with state-of-the-art methods

Table 4 provides a comparative analysis of the suggested BFDL model versus a number of other state-of-the-art algorithms such as a centralized Bi-LSTM, FedAvg, and FedProx. The centralised Bi-LSTM model scores 91.2% accuracy, with 90.4%, 90.9%, and 90.6% of precision, recall, and F1-score, respectively. Although the results produced by the model are rather impressive, they do not make use of privacy protection that is possible with the help of federated learning. The next model used for comparison, called FedAvg (Bi-LSTM), performs better than the previous one with respect to all criteria with 93.5% of accuracy, precision of 93.0%, recall of 92.8%, and F1-score of 92.9%. Such performance can be explained by the fact that fedAvg is capable of using the potential of decentralised models that aggregate knowledge and protect users' data at the same time. FedProx (Bi-LSTM) scores even higher with 94.2% of accuracy, 93.8% of precision, recall of 94.1%, and F1-score of 93.9%.

**Table 4.** Comparison with state of the arts methods.

| Model                  | Accuracy (%) | Precision (%) | Recall (%) | F1-Score (%) |
|------------------------|--------------|---------------|------------|--------------|
| Centralized Bi-LSTM    | 91.2         | 90.4          | 90.9       | 90.6         |
| FedAvg (Bi-LSTM) [28]  | 93.5         | 93.0          | 92.8       | 92.9         |
| FedProx (Bi-LSTM) [29] | 94.2         | 93.8          | 94.1       | 93.9         |
| Proposed BFDL          | 97.1         | 96.8          | 97.2       | 97.0         |

BFDL framework attains the highest performance with respect to all evaluation measures such as 97.1% for accuracy, 96.8% for precision, 97.2% for recall, 97.0% for the F1 measure, and 0.935 for the AUC value. The substantial improvement compared to other models, both centralized and federated, can be attributed to the use of blockchain to improve the secure aggregation process, along with trust-aware client selection and privacy-preserving techniques.

## 5. Practical Importance and Application Domains

The proposed Trust-Aware Federated Learning framework has practical relevance beyond conventional social media analysis because many modern applications require collaborative learning across distributed and privacy-sensitive data sources. In the social media sector, the framework can

support privacy-preserving detection of misinformation, disinformation, fake news, hate speech, cyberbullying, online harassment, and other harmful content without requiring the direct transfer of user-generated data. In healthcare, different hospitals and healthcare organizations could collaboratively develop models while retaining sensitive patient information within their respective environments. Similarly, financial institutions could use the framework for collaborative fraud detection, suspicious activity identification, and risk analysis while reducing the exposure of confidential customer information.

The framework is also applicable to cybersecurity environments, where organizations can collaboratively identify malicious activities and emerging threats without directly sharing sensitive security logs. In government and public-sector applications, it could support the analysis of misinformation and harmful online content while maintaining appropriate data governance and privacy controls. Furthermore, IoT and edge-computing environments can benefit from the framework because data is naturally distributed across numerous devices and edge nodes. In such environments, the trust score mechanism can help identify unreliable or potentially malicious participants before their model updates influence the global model. Therefore, the combination of federated learning, blockchain-based validation, trust-aware aggregation, and deep learning provides a practical architecture for secure and reliable collaborative AI across multiple privacy-sensitive sectors.

## 6. Conclusion

This research presents a trust-based federated learning framework that addresses key challenges in decentralized learning, particularly privacy preservation, participant reliability, and robustness against potentially adversarial behavior. The proposed framework achieves an accuracy of 97.1% and an AUC of 0.982, demonstrating its effectiveness in social media content classification and moderation. By keeping sensitive social media data within the local client environment and exchanging only model updates, the framework reduces direct exposure of user information while enabling collaborative model training.

The integration of the Trust Score Engine further strengthens the reliability of the federated learning process by identifying potentially unreliable or malicious clients and reducing their influence on global model aggregation. In addition, the lightweight blockchain layer provides transparent and verifiable management of model updates while maintaining efficient communication and validation. The observed latency and throughput results indicate that the framework can support increasing numbers of participating clients without substantial degradation in performance.

Despite these advantages, some limitations remain. The effectiveness of the framework depends on the accuracy of the trust-score calculation and the selected trust threshold. Moreover, variations in client data distributions and highly heterogeneous non-IID data may affect global model convergence. Although the blockchain architecture is designed to be lightweight, increasing the number of clients and model-update transactions may introduce additional computational, communication, and storage overhead. The current evaluation is also based on a specific dataset and experimental environment, and therefore further validation using larger and more diverse real-world datasets is required. Future work will focus on developing adaptive trust mechanisms, improving robustness against sophisticated adversarial behavior, handling greater client heterogeneity, and reducing blockchain-related overhead. Overall, the proposed framework provides a promising foundation for secure, privacy-preserving, trustworthy, and scalable decentralized social media analysis.

### Author Contributions

Conceptualization, methodology, software, formal analysis, investigation, data curation, and writing—original draft preparation, S.S.; Validation, visualization, data analysis, interpretation of results, and writing—review and editing, M.S.(Mridula Singh); Supervision, project administration, interpretation of results, writing—review and editing, and critical revision of the manuscript, M.S.(Manash Sarkar). All authors have read and agreed to the published version of the manuscript.

### Funding

This research received no external funding.

### Conflict of Interest Statement

The authors declare no conflicts of interest.

### Data Availability Statement

No new data were generated during the preparation of this manuscript.

## References

1. E. Aïmeur, S. Amri, and G. Brassard, "Fake news, disinformation and misinformation in social media: A review", *Social Network Analysis and Mining*, vol. 13(1), p. 30, 2023.
2. Saif, S., Islam, M. J., Jahangir, M. Z. B., Biswas, P., Rashid, A., Nasim, M. D., & Gupta, K. D. "A comprehensive review on understanding the decentralized and collaborative approach in machine learning", *arXiv preprint arXiv:2503.09833*, 2025.
3. Oyinloye, Oloyede, N. A. Oyegoke, V. E. Odion, and O. O. Ojewumi. "Regulation, censorship and media freedom", *African Journal of Social and Behavioural Sciences*, 14(2), pp. 651-670, 2024.
4. Wen, J., Zhang, Z., Lan, Y., Cui, Z., Cai, J. and Zhang, W. "A survey on federated learning: Challenges and applications", *International Journal of Machine Learning and Cybernetics*, 14(2), pp. 513-535, 2023.
5. Bajwa, M. T. T., Shafi, M. Z., Ur Rehman, M. A., Ali, A., Khawar, F., & Awais, M. "Blockchain-enabled federated learning for privacy-preserving AI application", *The Asian Bulletin of Big Data Management*, vol. 5, no. 3, pp. 154-169, 2025.
6. Abdullah, A., Ullah, F., Hafeez, N., Latif, I., Sidorov, G., Riveron, E. F., & Gelbukh, A. "Cyberbullying detection on social media using machine learning techniques", *Computación y Sistemas*, vol. 29, no. 3, 2025.
7. M. A. A. Abdullah, O. Kolesnikova, and G. Sidorov, "Detection of biased phrases in the Wiki Neutrality Corpus for fairer digital content management using artificial intelligence", *Big Data and Cognitive Computing*, 9(7), p. 190, 2025.
8. Mohasseb, A., Amer, E., Chiroma, F., & Tranchese, A. "Leveraging advanced NLP techniques and data augmentation to enhance online misogyny detection", *Applied Sciences*, 15(2), p. 856, 2025.
9. Liu, Z., Guo, J., Yang, W., Fan, J., Lam, K. Y., & Zhao, J. "Privacy-preserving aggregation in federated learning: A survey", *IEEE Transactions on Big Data*, 2022.
10. Xu, R., Baracaldo, N., Zhou, Y., Anwar, A., Joshi, J. and Ludwig, H. "FedV: Privacy-preserving federated learning over vertically partitioned data," in *Proc. 14th ACM Workshop on Artificial Intelligence and Security*, pp. 181-192, 2021.
11. Fereidooni, H., Marchal, S., Miettinen, M., Mirhoseini, A., Möllering, H., Nguyen, T.D., Rieger, P., Sadeghi, A.R., Schneider, T., Yalame, H. and Zeitouni, S. "SAFElearn: Secure aggregation for private federated learning", in *Proc. IEEE Security and Privacy Workshops (SPW)*, pp. 56-62, 2021.
12. Chen, C., Liu, J., Tan, H., Li, X., Wang, K.I.K., Li, P., Sakurai, K. and Dou, D. "Trustworthy federated learning: Privacy, security, and beyond", *Knowledge and Information Systems*, 67(3), pp. 2321-2356, 2025.
13. Mahato, G.K., Banerjee, A., Chakraborty, S.K. and Gao, X.Z. "Privacy preserving verifiable federated learning scheme using blockchain and homomorphic encryption", *Applied Soft Computing*, 167, p. 112405, 2024.
14. Nguyen, D.C., Ding, M., Pham, Q.V., Pathirana, P.N., Le, L.B., Seneviratne, A., Li, J., Niyato, D. and Poor, H.V. "Federated learning meets blockchain in edge computing: Opportunities and challenges", *IEEE Internet of Things Journal*, 8(16), pp. 12806-12825, 2021.
15. Kalapaaking, A. P., Khalil, I., Rahman, M. S., Atiquzzaman, M., Yi, X., & Almashor, M. "Blockchain-based federated learning with secure aggregation in trusted execution environment for internet-of-things", *IEEE Transactions on Industrial Informatics*, 19(2), pp. 1703-1714, 2022.
16. Nazir, A., He, J., Zhu, N., Anwar, M.S. and Pathan, M.S. "Enhancing IoT security: A collaborative framework integrating federated learning, dense neural networks, and blockchain", *Cluster Computing*, 27(6), pp. 8367-8392, 2024.
17. S. Kridera and A. Kanavos, "Exploring trust dynamics in online social networks: A social network analysis perspective", *Mathematical and Computational Applications*, 29(3), p. 37, 2024.
18. Ahmed, A., Saleem, K., Rashid, U. and Baz, A. "Modeling trust-aware recommendations with temporal dynamics in social networks", *IEEE Access*, 8, pp. 149676-149705, 2020.

19. Chen, J., Yan, H., Liu, Z., Zhang, M., Xiong, H., & Yu, S. "When federated learning meets privacy-preserving computation", *ACM Computing Surveys*, 56(12), pp. 1-36, 2024.
20. Rawat, H. and Rajavat, A. "Trust-Aware Federated User Representation Learning for Privacy-Preserving and Carbon-Optimized Personalized Recommendation in Large-Scale Social Platforms", *Human-Centric Intelligent Systems*, 6(2), pp. 219-243, 2026.
21. Singh, S., Singh M., and Sarkar M. "Secure Data Transaction Under Social Network: An Inspiring Influence of Members", In *2026 International Conference on Innovations in Computational Intelligence (ICICI)*, pp. 1-7. IEEE, 2026.
22. EL-Sayed, Tharwat, Ayman El-Sayed, and Abdullah N. Moustafa. "Tweet tone triage technique (4T): a secured federated deep learning approach", *Neural Computing and Applications*, 38(10), p. 364, 2026.
23. Sunny, Md Arif Istiaque, Mirza Mohd Shahriar Maswood, and Abdullah G. Alharbi. "Deep learning-based stock price prediction using LSTM and bi-directional LSTM model", In *2020 2nd novel intelligent and leading emerging sciences conference (NILES)*, pp. 87-92. IEEE, 2020.
24. F. Yang, W. Zhou, Q. Wu, R. Long, N. N. Xiong, and M. Zhou, "Delegated proof of stake with downgrade: A secure and efficient blockchain consensus algorithm with downgrade mechanism", *IEEE Access*, 7, pp. 118541–118555, 2019.
25. A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains", in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, pp. 3–16, 2016.
26. F. Saleh, "Blockchain without waste: Proof-of-stake", *Rev. Financial Stud.*, 34(3), pp. 1156–1190, 2021.
27. H. Xia, "Continuous-bag-of-words and Skip-gram for word vector training and text classification", *J. Phys.: Conf. Ser.*, 2634(1), p. 012052, 2023.
28. B. Casella and S. Fonio, "Architecture-based FedAvg for vertical federated learning", in *Proc. IEEE/ACM 16th Int. Conf. Utility Cloud Comput.*, pp. 1–6, 2023.
29. T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks", *Proc. Mach. Learn. Syst.*, 2, pp. 429–450, 2020.
30. T. Davidson, D. Warmley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language", in *Proc. Int. AAAI Conf. Web Soc. Media*, 11(1), pp. 512–515, 2017.