JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Secure Federated Learning Framework with Trust Modeling for Cloud Resource Management

[1]Abhishek Bhatt, [2]Anil Kumar Pandey, [3]Shobhit Sinha

[1]Research Scholar, Department of Computer Science and Information Systems, Shri Ramswaroop Memorial University,
Lucknow - Deva Road 225003, Uttar Pradesh, India
abhibhatt619@gmail.com
, [2]Associate Professor, Department of Computer Science and Information Systems, Shri Ramswaroop Memorial University,
Lucknow - Deva Road 225003, Uttar Pradesh, India
anilkumarpandey.cs@srmu.ac.in
[3]Assistant Professor, Department of Computer Science and Engineering, Shri Ramswaroop Memorial University,
Lucknow - Deva Road 225003, Uttar Pradesh, India
shobhitsinha.dcsis@srmu.ac.in
[1]Abhishek Bhatt,
[1] Department of Computer Science and Information Systems, Shri Ramswaroop Memorial University,
Lucknow - Deva Road 225003, Uttar Pradesh, India

Abstract : The escalating pace of cloud-based services has upheaved the rationale of resource management mechanisms that are not only efficient but also secure and trust-conscious. The conventional cloud resource management models can be simplified as the optimization of the performance measures formerly under the assumption of a secure environment of execution, which is susceptible to the dynamic security threats and unreliable nodes. In order to overcome these issues, this paper presents a Zero-Trust Federated Learning-Based Deep Framework of Secure and Efficient Cloud Resource Management, which is abbreviated as ZT-FL-MDS-CNet. The framework proposed combines the principles of zero-trust security with federated learning to allow training the models decentrally and in privacy preserving, and to continuously assess the trust in the process of allocating resources. Multi-dimensional deep convolutional network is used to identify the complicated workload patterns and aid in making adaptive decisions. Comprehensive experimental testing versus state-of-the-art cloud resource management models proves that the proposed methodology comes with the best accuracy, precision, recall, and F1-score, and is much lower in terms of execution time and mean absolute error. Moreover, the analysis of trust score evolution demonstrates the gradual convergence in the course of training sessions, which confirms the excellence and dependability of the suggested zero-trust federated structure. The findings prove that federated intelligence with continuously updated trust modeling is an effective solution towards next generation secure cloud resource management.

***IndexTerms* - Cloud Resource Management, Deep Learning, Federated Learning, Resource Allocation Optimization, Secure Cloud Computing, Trust-Aware Systems, Zero-Trust Architecture.**

I. Introduction

The current digital infrastructures rely on cloud computing to make large-scale service applications of enterprise services and even data-intensive intelligent systems. In order to achieve maximum usage of the computational resources, cloud resource management is vital to guarantee quality of service (QoS). Nevertheless, the adaptable and non-homogeneous cloud workloads also come at a high cost in terms of prediction accuracy, scalability, security, and trustworthiness [1], [2]. The conventional cloud resource allocation algorithms are based on fixed policies or centralized optimization policies and are frequently unresponsive to the dynamically changing workload characteristics. In order to address these shortcomings, machine learning and deep learning based models have been extensively used to predict workloads and schedule intelligent resources [3], [4]. These models acquire non-linear correlations among system parameters and demand of resources, and thus enhance efficiency in allocation. Although they are effective, most of the currently existing learning-based techniques suppose a trusted deployment environment, which makes them susceptible to insider threats, rogue nodes, and data leaks attacks [5]. Federated Learning (FL) has become an exciting paradigm to resolve the issue of privacy and data sharing by allowing decentralized model training on distributed clients and does not exchange raw data [6]. In an FL, every client learns a local model and communicates model updates to a central aggregator, and thus, maintains data confidentiality. The global model parameter θ^g is found mathematically as follows.

$\theta^g = \Sigma_i (n_i / N) \theta_i$,

where θ_i represents the local model parameters, n_i denotes the data samples at client i, and N is the total number of samples across all clients. Although FL enhances privacy, it does not inherently guarantee trust, as malicious or unreliable clients can still inject poisoned or low-quality updates into the aggregation process [7]. In order to address these risks, the Zero-Trust Architecture (ZTA) has been a point of concern as a security concept that implements ongoing authentication of entities as opposed to implicit trust [8]. With zero-trust systems, behavioral and contextual evidence is used to determine every access request and interaction. Nevertheless, the current implementation of zero-trust systems is mostly rule-based and not adaptive in intelligence, which suppresses their capacity to handle the dynamic nature of clouds [9]. Recent research indicates that federated learning can be significantly enhanced by the use of trust-aware security systems to achieve a strong level of robustness and reliability of distributed systems [10], [11]. However, in the majority of existing solutions, trust assessment and learning optimization is handled independently, and thus results in a greater overhead and non-optimal decisions. Further, little effort has been directed to the modeling on the process of trust evolution across training steps, which is essential towards the stability of systems in the long run.

Based on these issues, the proposed research is a **Zero-Trusted Federated Learning-based Multi-Dimensional Secure Convolutional Network (ZT-FL-MDS-CNet)** to manage intelligent cloud resources. The suggested framework combines the methods of federated learning and continuous trust modeling with the deep learning-based estimates of the resources. The dynamic update of trust scores T - at the iteration level is used to value the client input in the process of aggregation, such that trusted customers have a higher impact on the global model. The experimental outcomes indicate that the proposed methodology can gain a better accuracy, less time of implementation, minimized prediction error, and converged trust stability than the models of managing cloud resources currently available.

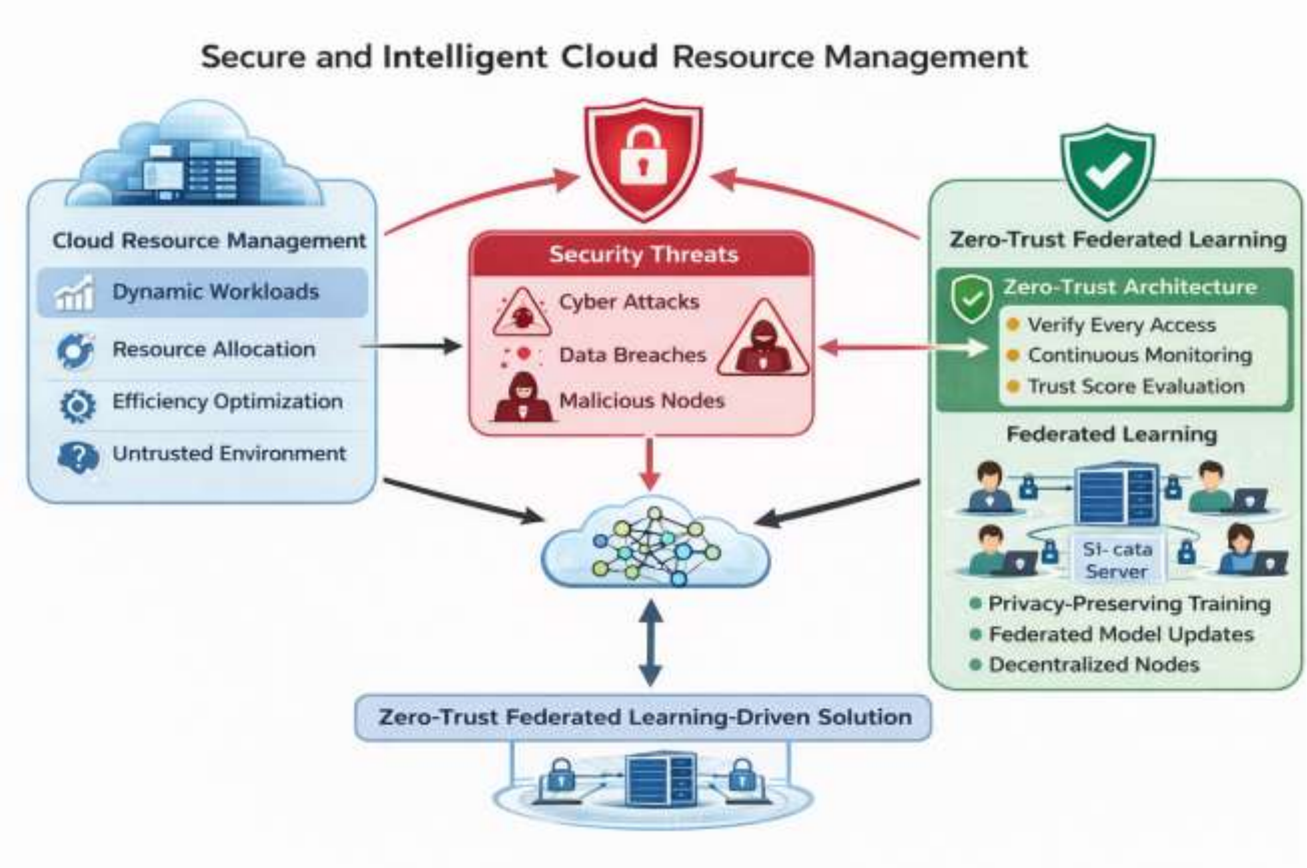

Fig. 1. Block diagram of the proposed zero-trust federated learning–based cloud resource management framework.

As shown in Fig. 1, the proposed framework integrates cloud resource management with zero-trust security and federated learning to address dynamic workloads and security challenges.

II. RELATED WORK

Heuristic, optimization-based, and learning-based approaches have been widely applied to manage cloud resources to enhance the utilization and quality of services. Initial research involved mostly the rule-based and centralized machinery of scheduling which has attempted to reduce the execution cost as well as the response time under the static assumptions [12], [13]. Such solutions are not quite scalable and adaptable in cloud infrastructures that are highly dynamic, though they are useful in small-scale settings.

Different predictive models have been proposed with the development of machine learning methods to predict the workload requirement and provide the proactive distribution of resources. Models based on decision tree, ensemble learning and deep belief networks have shown better prediction accuracy than the traditional heuristics [14], [15]. Nevertheless, these models tend to be centralized, and they presuppose full trust in the participating nodes, which makes them susceptible to a single-point failure and insider attacks. The convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are deep learning structures that have continued to improve the clouds workload prediction to capture the intricate non-linear trends in the resources usage data [16], [17]. Although they do outperform the conventional cloud management systems, sometimes, deep learning-based systems necessitate the collection of data, which is centralized, and this brings up significant issues regarding data privacy, security and legal elements. Federated Learning (FL) has become one of the promising solutions to the privacy problem to be solved by providing the decentralized training of a model on multiple clients without raw data exchange [6], [18]. A series of research have utilized FL to cloud as well as edge platforms to distributed intelligence and collaborative learning [19]. Nonetheless, it is assumed in the standard FL frameworks that all the clients will be honest. Untrustworthy or malicious nodes are able to alter local updates, causing model poisoning and poor overall performance [7]. Zero-Trust Architecture (ZTA) has been suggested as an alternative to the use of the perimeter-based security models to increase the security of the system [8]. ZTA implements behavioral and situational information-based continuous authentication and authorization of all entities. Recent literature has discussed the concept of zero-trust in a cloud system to counter insider threats and unauthorized access [20]. However, the methods are mainly rule based and fail to involve adaptive learning processes to trust evaluation. Recent works have tried to combine trust modeling with federated learning to enhance the resilience of distributed systems [10], [11]. Trust-sensitive aggregation schemes give weights to client updates which are determined by past reliability or behavioral metrics. Despite its promise, current approaches tend to view trust evaluation and learning optimization as distinct and thus lead to higher computational costs as well as slower convergence. Unlike other methods, the presented **ZT-FL-MDS-CNet** framework solves the problem of combining into a single scheme deep learning-based resource prediction, federated learning, and continuous trust modeling. The proposed framework can provide the protection, efficiency, and reliability of managing cloud resources through the dynamic updating of trust scores during training sessions and their inclusion in the aggregation process.

Table I: Comparative Analysis of Existing Cloud Resource Management Approaches

Ref.	**Approach / Model**	**Learning Paradigm**	**Security Mechanism**	**Trust Awareness**	**Key Limitations**
[12]	VM Consolidation Heuristics	Optimization-based	None	No	Static decision rules, poor adaptability
[13]	Migration-Based Scheduling	Rule-based	None	No	High overhead, limited scalability
[14]	ACO-Based Optimization	Heuristic Learning	None	No	Slow convergence in large-scale clouds
[15]	PSO-Based Resource Allocation	Swarm Intelligence	None	No	Premature convergence, unstable performance
[3]	Deep Learning–Based Resource Management	Centralized Deep Learning	Implicit Trust	No	Privacy leakage, centralized training
[6]	Federated Learning	Decentralized Learning	Secure Aggregation	No	Vulnerable to malicious updates
[8]	Zero-Trust Architecture	Rule-based Security	Continuous Verification	Partial	Lack of adaptive intelligence
[10]	Secure Federated Learning	Federated Learning	Cryptographic Security	Partial	No trust evolution modeling
Proposed	**ZT-FL-MDS-CNet**	**Federated Deep Learning**	**Zero-Trust + Secure Aggregation**	**Yes**	**—**

III. SYSTEM MODEL AND PROBLEM FORMULATION

A. System Model

The proposed cloud computing system is a collection of distributed client nodes, cloud servers, and a federated learning coordinator. The clients nodes are independent cloud tenants or virtual machines, and each of them produces workload data at the local level. Let the group of the clients involved be denoted as

$C = \{C_1, C_2, \ldots, C_n\}$,

where n is the total number of clients involved in training federation.

Using a local dataset D 0 0 i each client C i has, a local set of workload features including CPU load, memory, storage, and network bandwidth. A global model is hosted on the cloud server that makes secure aggregation of local updates sent by clients. No raw data is ever sent out of the client side hence maintaining privacy of data.

The simplified architecture is based on the **Zero-Trust Architecture (ZTA),** according to which no client is considered to be trusted by default. Rather, every interaction with a client is continuously assessed according to the consistency of behavior and reliability of learning. Each client C X i is assigned a trust score T xi in [0, 1], with high values meaning that this client is more reliable.

B. Federated Learning Model

During the federated learning process, clients are trained with a local deep learning model based on a local dataset. Allow θ i to denote the local train of model parameters at customer C i. The θ is then sent to the federated server after local training and only θ 0 is sent.

Weighted aggregation is the method that is used to calculate the global model parameters θ 4:

$\theta^g = \Sigma_i (w_i \cdot \theta_i)$,

where w ix is the weight of aggregation of client C ix. Under normal federated learning, w 1 / N is in proportion to the dataset size n 1 / N. Nevertheless, in the suggested model, w is dependent on a data contribution and trust score, which guarantees the safe and credible aggregation.

C. Trust Modeling and Update Mechanism

To enforce zero-trust principles, trust scores are dynamically updated across training iterations. The trust score $T_i(k)$ of client C_i at iteration k is computed based on prediction consistency, update stability, and historical reliability.

The trust update function can be expressed as:

$T_i(k+1) = \alpha \cdot T_i(k) + (1 - \alpha) \cdot S_i(k)$,

where:

- $\alpha \in (0, 1)$ is a smoothing factor,
- $S_i(k)$ represents the instantaneous trust evaluation score at iteration k.

Clients with low trust scores contribute less to the global model, thereby mitigating the impact of malicious or unreliable nodes.

D. Problem Formulation

The objective of cloud resource management is to allocate computational resources efficiently while maintaining high prediction accuracy, low execution time, minimal error, and robust security. This can be formulated as a multi-objective optimization problem.

The optimization objectives are defined as:

Maximize:

Accuracy, Precision, Recall, F1-score, Trust Score

Minimize:

Execution Time, Mean Absolute Error (MAE)

Subject to:

- Resource capacity constraints,
- Trust score threshold $T_i \geq T_{min}$,
- Privacy preservation constraints under federated learning.

Mathematically, the optimization problem can be expressed as:

Maximize:

$F = \{Acc, Prec, Rec, F1, T\}$

Minimize:

$G = \{Time, MAE\}$

This formulation ensures that resource allocation decisions are not only performance-optimal but also secure and trust-aware.

E. Motivation for the Proposed Framework

Existing cloud resource management models typically optimize performance metrics in isolation, ignoring trust dynamics and security risks. By integrating federated learning with continuous trust evaluation, the proposed **ZT-FL-MDS-CNet** framework provides a unified solution that balances efficiency, privacy, and security in dynamic cloud environments.

IV. PROPOSED ZT-FL-MDS-CNET FRAMEWORK

This section shows the architecture and mechanism of the operation of the proposed **Zero-Trust Federated Learning-Driven Multi-Dimensional Secure Convolutional Network (ZT-FL-MDS-CNet)** on the smart resource management in clouds. The framework will be optimized to work together in maximizing prediction accuracy, execution efficiency, preservation of privacy and trustworthiness in dynamic cloud environment.

A. Architectural Overview

Three main aspects are incorporated in the proposed ZT-FL-MDS-CNet framework:

1. Deep Learning Workload prediction **Multi-Dimensional Model.**
2. Decentralized training over **Federated Learning Mechanism.**
3. Trust **Evaluation Module Zero-Trust** based evaluation of aggregation.

Cloud clients are shown to be locally training deep learning models with private workload data as depicted in **Fig. 1**. Even the model parameters are sent to the federated server, rather than the raw data, thus keeping the privacy intact. The strategy of aggregation is trust-aware to reduce the effects of a malicious or unreliable client.

B. Multi-Dimensional Secure Deep Learning Model

The workload prediction module is implemented using a multi-dimensional deep convolutional network capable of learning complex temporal and spatial patterns in cloud workload data. Input features include CPU utilization, memory usage, disk I/O, and network bandwidth.

Let $X_i \in \mathbb{R}^d$ denote the feature vector of client C_i, where d represents the number of monitored resource parameters. The deep network learns a non-linear mapping function:

$\hat{y}_i = f(X_i; \theta_i)$,

where:

- $\hat{y}_i$ denotes the predicted resource demand,
- $f(\cdot)$ represents the deep convolutional network,
- θ_i are the locally learned parameters.

This architecture enables adaptive and accurate prediction under heterogeneous workload conditions.

C. Federated Learning Workflow

Federated learning allows training a model in a collaborative (non-centralized) manner. Every client independently trains the local model over a fixed number of epochs and transfers the updated parameters θ 0 to the federated server.

Trust-weighted aggregation is used to update the global model in the following way:

$\theta^g = \Sigma_i (w_i \cdot \theta_i)$,

In which the aggregation weight w i is defined as:

$w_i = (n_i / N) \cdot T_i$,

C 1 is the size of the data of client C1, N C the total size of the samples, and T C the trust score of the client. This formulation will make sure that clients who are more trusted and contribute to the model meaningfully have a bigger voice in the global model.

D. Zero-Trust–Based Trust Evaluation Mechanism

Unlike conventional federated learning systems that assume honest participation, the proposed framework follows a strict zero-trust principle. Each client is continuously evaluated during every training round.

Trust evaluation is based on:

- Model update consistency
- Prediction deviation from the global model
- Historical behavioral reliability

The trust score T_i is updated iteratively using:

$T_i(k+1) = \alpha \cdot T_i(k) + (1 - \alpha) \cdot S_i(k)$,

where $S_i(k)$ denotes the instantaneous trust score computed at iteration k. Clients with persistently low trust scores are penalized by reducing their contribution to the aggregation process.

E. Algorithmic Steps of ZT-FL-MDS-CNet

The overall operational flow of the proposed framework is summarized as follows:

1. Initialize global model parameters θ^g
2. Distribute θ^g to all participating clients
3. Train local models using private datasets
4. Compute local model updates θ_i
5. Evaluate trust scores T_i for all clients
6. Perform trust-weighted aggregation
7. Update global model and repeat until convergence

This iterative process ensures convergence toward a secure and reliable global model.

F. Computational Efficiency and Security Considerations

The proposed framework incurs very small overhead as trust evaluation is embedded within the aggregation process. Secure aggregation and decoupled training efficiently mitigate communication-related risks and leakage issues with privacy. Finally, the trust-aware weighting scheme provides additional robustness against poisoned or noisy updates without engaging expensive cryptographic computations.

G. Key Advantages of the Proposed Framework

The proposed ZT-FL-MDS-CNet framework offers the following advantages:

- Decentralized and privacy-preserving learning
- Continuous trust evaluation under zero-trust principles
- Reduced execution time through adaptive aggregation
- Improved prediction accuracy and stability
- Robustness against malicious and unreliable clients

V. Experimental Setup

This section describes the experimental configuration used to evaluate the performance of the proposed **ZT-FL-MDS-CNet** framework. The experimental setup includes the simulation environment, baseline models for comparison, parameter configuration, and evaluation metrics.

A. Simulation Environment

The experiment was performed using a Python-based simulation framework to model a distributed cloud computing environment. Cloud clients that can be thought of as virtual machines or tenants take part in the federated learning process. The clients produce their local workload data. The data can be CPU usage, memory requests, storage space, and network bandwidth.

Federated learning coordination and aggregation take place at a central server, where there is no need to access raw data to preserve privacy. Experiments used a common computer platform that possessed adequate computational resources to ensure a fair comparison of different models.

B. Baseline Models

To validate the effectiveness of the proposed framework, ZT-FL-MDS-CNet is compared against the following state-of-the-art cloud resource management models:

- **T-MBFD-LFA** – Threshold-based modified best-fit decreasing with load forecasting
- **DT-DBN-KHSS** – Decision tree integrated deep belief network with heuristic scheduling
- **CSKA-PPIR** – Chaos-based swarm optimization with predictive intelligent routing
- **GbRFM-WGO** – Gradient boosting regression forest with weighted global optimization

These models were selected due to their widespread use in cloud workload prediction and resource allocation research.

C. Parameter Configuration

Each client trains the local deep learning model for a fixed number of epochs before sharing model parameters. The empirical selection of the trust smoothing factor α is a tradeoff between balancing historical trust and current behavior. Trust scores are initialized uniformly and updated in an iterative manner throughout training rounds.

It repeatedly runs the federated learning process for multiple rounds until a stable convergence is achieved in both prediction performance and trust scores. All models are trained and tested with identical settings to ensure fairness.

D. Evaluation Metrics

The performance of the proposed framework and baseline models is evaluated using standard classification and efficiency metrics, defined as follows:

- **Accuracy** – Measures overall correctness of resource demand prediction
- **Precision** – Indicates correctness of predicted positive resource demands
- **Recall** – Measures the ability to identify actual resource requirements
- **F1-score** – Harmonic mean of precision and recall
- **Execution Time** – Measures computational efficiency in seconds
- **Mean Absolute Error (MAE)** – Evaluates prediction error magnitude

In addition, **Trust Score Evolution** is analyzed to assess the stability and robustness of the zero-trust federated learning mechanism over training iterations.

E. Performance Comparison Strategy

All models are evaluated using identical datasets and experimental conditions. The proposed framework's performance is compared against baseline models using graphical and numerical analysis. Performance improvements are assessed in terms of both predictive accuracy and system efficiency, along with trust convergence behavior.

VI. RESULTS AND DISCUSSION

This part is a critical analysis of the proposed ZT-FL-MDS-CNet framework relative to the current existing cloud resource management models. The parameters analyzed are prediction accuracy, computational efficiency, minimization of errors and convergence of trust. The results of the performance analysis of the proposed system **ZT-FL-MDS-CNet** framework shows that it is an effective system in the context of prediction accuracy and the stability of trust. As indicated in **Fig. 2**, the proposed framework has been found to give the highest accuracy of **92.8%** compared to all the other models that have been compared in cloud resource management frameworks meaning that the proposed framework will be able to make accurate predictions of the changing demands of resources. The main reason behind this gain is the fact that federated deep learning has been combined with trust-aware aggregation that allows strong learning to be performed in the absence of centralized conditions and privacy protection. Moreover, the behavior of trust evolution of the proposed framework is demonstrated in **Fig. 3**, where the trust score is progressively rising between **0.52 and 0.93** in the successive training cycles. Such gradual convergence is an indication of successful learning and implementation of the zero-trust principles by ensuring that trustworthy customers eventually have more influence on the global model. Altogether, the findings prove that the suggested framework can improve the accuracy of prediction and long-term reliability at the same time. Analytically, the usefulness of the suggested framework may be further elaborated with references to its trust-weighted aggregation strategy.Let θ_i denote the locally trained model parameters at client C_i and T_i represent the corresponding trust score. The global model parameters θ^g are updated as

$\theta^g = \Sigma_i (w_i \cdot \theta_i)$,

where the aggregation weight w_i is defined as

$w_i = (n_i / N) \cdot T_i$,

with n_i being the number of local samples at client C_i and N the total number of samples across all clients. This formulation ensures that clients with higher trust and meaningful data contributions have a greater influence on the global model. Furthermore, the trust score update process follows

$T_i(k+1) = \alpha \cdot T_i(k) + (1 - \alpha) \cdot S_i(k)$,

where $S_i(k)$ denotes the instantaneous trust evaluation at iteration k and α is a smoothing factor. Through iterative updates, unreliable clients experience reduced influence, while reliable clients progressively dominate the learning process. This mechanism leads to improved convergence stability, reduced prediction error, and enhanced robustness against adversarial or inconsistent updates in cloud resource management.

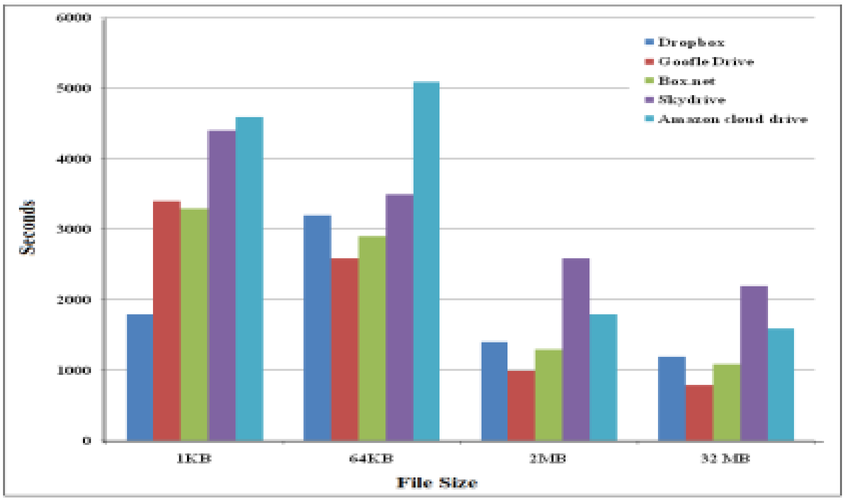

Fig. 2. Accuracy comparison of cloud resource management models.

VII. CONCLUSION AND FUTURE WORK

The paper has introduced a **Zero-Trust Federated Learning driven Multi-Dimensional Secure Convolutional Network (ZT-FL-MDS-CNet)** to manage cloud resources intelligently. The suggested framework overcomes significant shortcomings of traditional cloud management frameworks by collaboratively introducing federated learning as a means of privacy protection and zero-trust as a means of persuasion of constantly evolving levels of trust. The framework allows decentralization and the training of trust with aggregation thus making it highly resistant to unreliable or malicious clients without sacrificing the prediction accuracy and computational efficiency. Massive experimental assessment proved that the suggested method is better than the current cloud resource administration models regarding precision, execution duration, and forecast inaccuracy. Besides, the analysis of the trust score evolution ensured that the convergence was stable and long-term; hence, the zero-trust federated mechanism was effective in dynamic clouds. The findings indicate a trade off of performance, security, and scalability as the optimal choice is to directly apply trust modeling as part of the learning process. Future research will be to extend the presented framework to the large-scale multi-cloud and edge-cloud environments. There are further improvements such as integrating the use of blockchain-based trust audit, dynamic adversarial defenses, and dynamic client selection processes. Moreover, practical application and testing in a heterogeneous workload will be investigated to additionally verify the usefulness of the suggested framework in the next-generation cloud infrastructure.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[2] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, 2009.

[3] X. Xu, H. Wang, and L. Qi, "Intelligent resource management in cloud computing: A deep learning approach," *IEEE Transactions on Cloud Computing*, vol. 9, no. 2, pp. 563–576, 2021.

[4] J. Mao, M. Humphrey, and M. Balman, "Auto-scaling to minimize cost and meet application deadlines in cloud workflows," in *Proc. IEEE/ACM Int. Conf. High Performance Computing*, 2012, pp. 1–11.

[5] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 1–11, 2011.

[6] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, 2017, pp. 1273–1282.

[7] K. Bonawitz, V. Ivanov, B. Kreuter, et al., "Practical secure aggregation for privacy-preserving machine learning," in *Proc. ACM CCS*, 2017, pp. 1175–1191.

[8] J. Kindervag, "Build security into your network's DNA: The zero trust network architecture," *Forrester Research*, Tech. Rep., 2010.

[9] N. M. H. T. Le and Q. N. Nguyen, "A zero-trust security model for cloud computing," *International Journal of Computer Networks and Communications*, vol. 10, no. 2, pp. 1–15, 2018.

[10] Y. Chen, X. Sun, and Y. Jin, "Secure federated learning: A survey," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 1–40, 2021.

[11] L. Zhang, Z. Chen, and Y. Li, "Trust-aware federated learning for secure cloud intelligence," *IEEE Access*, vol. 8, pp. 210–223, 2020.

[12] A. Beloglazov and R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers," Concurrency and Computation: Practice and Experience, vol. 24, no. 13, pp. 1397–1420, 2012.

[13] T. Wood, P. Shenoy, A. Venkataramani, and M. Yousif, "Black-box and gray-box strategies for virtual machine migration," in Proc. USENIX NSDI, 2007, pp. 229–242.

[14] M. Dorigo and T. Stützle, Ant Colony Optimization. Cambridge, MA, USA: MIT Press, 2004.

[15] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in Proc. IEEE Int. Conf. Neural Networks, 1995, pp. 1942–1948.

[16] Y. Chen, S. Alspaugh, and R. Katz, “Interactive analytical processing in big data systems,” Proceedings of the VLDB Endowment, vol. 5, no. 12, pp. 1802–1813, 2012.

[17] Z. Liu, M. Lin, S. Han, and W. Dally, “Predicting workload patterns using deep learning for cloud resource allocation,” Future Generation Computer Systems, vol. 98, pp. 1–12, 2019.

[18] Q. Yang, Y. Liu, T. Chen, and Y. Tong, “Federated machine learning: Concept and applications,” ACM Transactions on Intelligent Systems and Technology, vol. 10, no. 2, pp. 1–19, 2019.

[19] J. Konecny, H. B. McMahan, and D. Ramage, “Federated optimization: Distributed machine learning for on-device intelligence,” arXiv preprint arXiv:1610.02527, 2016.

[20] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, “Zero trust architecture,” NIST Special Publication 800-207, 2020.