TrustGraph-DFL: Byzantine-Resilient Decentralized Federated Learning via Consistency-Weighted Neighborhood Aggregation

Rao Xu , Yun Yang , Jiarong Qiu , Hengguang Cui , Yilin Sun , Zhongkang Li *

Posted Date: 15 April 2026

doi: 10.20944/preprints202604.1102.v1

Keywords: index terms-decentralized federated learning; byzantine resilience; trust graph; robust aggregation; poisoning attacks

 doi:10.20944/preprints202604.1102.v1

Article

TrustGraph-DFL: Byzantine-Resilient Decentralized Federated Learning via Consistency-Weighted Neighborhood Aggregation

Rao Xu [1], Yun Yang [2], Jiarong Qiu [3], Hengguang Cui [4], Yilin Sun [5] and Zhongkang Li [6,*]

[1] University of Toledo, Toledo, USA
[2] Northeastern University, Boston, USA
[3] University of Southern California, Los Angeles, USA
[4] Brown University, Providence, USA
[5] University of Pennsylvania, Philadelphia, USA
[6] New York University, New York, USA
* Correspondence: zhongkang0208@gmail.com

Abstract

Decentralized federated learning (DFL) eliminates the single point of failure inherent in server-based architectures, enabling peer-to-peer collaborative model training. However, the absence of a central authority makes DFL particularly vulnerable to Byzantine attacks from malicious participants. Existing Byzantine-robust methods often fail to exploit the network topology structure of DFL. We propose TrustGraph-DFL, a novel defense mechanism that leverages graph-based trust modeling for Byzantine resilience. Our key insight is that consistency between a neighbor's model update direction and a node's local validation gradient can serve as an effective trust indicator. Each node computes consistency scores by comparing received updates against locally computed validation gradients, then maps these scores to dynamic edge weights for robust weighted aggregation. Experiments on CIFAR-10 demonstrate that TrustGraph-DFL achieves 3--5% higher accuracy than existing methods under 30% Byzantine nodes while maintaining a low false positive rate (approximately 9% at 50% Byzantine fraction, compared to 35% for Krum).

Keywords: index terms-decentralized federated learning; byzantine resilience; trust graph; robust aggregation; poisoning attacks

1. Introduction

Federated learning (FL) has emerged as a prominent paradigm for privacy-preserving distributed machine learning, allowing multiple participants to collaboratively train a shared model without exchanging raw data [1]. In conventional FL, a central server coordinates the training process by aggregating model updates from clients. However, this server-assisted architecture introduces scalability bottlenecks and creates a single point of failure that can be exploited by adversaries.

Decentralized federated learning (DFL) addresses these limitations by enabling peer-to-peer model exchange without requiring a central aggregator [2]. In DFL, each participant communicates only with its direct neighbors in a network graph, aggregating received models locally before the next training round. This architecture naturally distributes trust and eliminates centralized vulnerabilities, making it attractive for applications in autonomous vehicles, healthcare networks, and edge computing [3].

Despite these advantages, DFL introduces new security challenges. The lack of a trusted central coordinator means that any participant can potentially inject malicious updates that propagate through

 doi:10.20944/preprints202604.1102.v1

the network. Byzantine attacks, where compromised nodes send arbitrary or carefully crafted malicious updates, pose a particularly severe threat [4]. These attacks can significantly degrade model performance or even cause complete training failure.

Several Byzantine-robust aggregation rules have been proposed for centralized FL, including Krum [4], trimmed mean [5], and trust-based methods like FLTrust [6]. However, directly applying these methods to DFL is problematic for two reasons. First, they assume access to all client updates at a central point, which is unavailable in DFL. Second, they fail to exploit the network topology structure that is fundamental to DFL communication patterns. Recent works have begun addressing Byzantine resilience specifically for DFL. BALANCE [7] proposes local similarity-based filtering where each node uses its own model as a reference. DFL-Dual [8] employs dual-domain clustering to separate benign and Byzantine clients. While these methods represent significant progress, they either require additional communication rounds or assume specific attack models.

In this paper, we propose TrustGraph-DFL, a novel Byzantine-resilient framework that naturally integrates trust assessment into the decentralized aggregation process. Our approach builds on a key observation: when a neighbor's model update direction aligns with a node's local validation gradient, this consistency suggests benign behavior. Conversely, malicious updates typically exhibit low or negative consistency with honest nodes' learning objectives.

Our main contributions are summarized as follows:

- We introduce a consistency-based trust scoring mechanism that computes the directional alignment between received neighbor updates and locally computed validation gradients.
- We propose a dynamic trust graph where edge weights reflect consistency scores, enabling topology-aware robust aggregation that naturally down-weights suspicious contributions.
- We conduct comprehensive experiments under three attack types across three network topologies, demonstrating consistent improvements over existing methods.

2. Methodology Foundations of the Proposed Approach

The methodological foundation of TrustGraph-DFL is established around a central problem in decentralized federated learning: robust peer-to-peer model aggregation must be achieved without a central coordinator, while adversarial participants may exploit the distributed topology to inject misleading updates. From this perspective, the proposed trust-graph formulation is not an isolated design choice, but a synthesis of several methodological strands represented in the reference set. First, graph-structured modeling provides the most direct formal basis for representing decentralized interaction patterns. Structure-aware semantic graphs and graph-enhanced anomaly modeling demonstrate that relational structure itself can be used as a discriminative signal rather than merely as a communication constraint [9]. Multi-hop relational reasoning further shows that graph connectivity can reveal higher-order behavioral dependencies that are difficult to capture through pointwise update comparison alone [10]. Adaptive graph construction with spatiotemporal contrastive objectives extends this idea by suggesting that the graph should not remain static, but should evolve with observed behavior and consistency patterns [11]. Related graph-based contrastive learning for performance anomaly prediction and dynamic spatiotemporal causal graph neural modeling reinforce the value of encoding both local relations and temporal evolution into graph-based decision mechanisms [12,13]. These works collectively justify the core methodological premise of TrustGraph-DFL: the communication topology in DFL should be transformed into a trust-sensitive graph whose edge weights are dynamically modulated by observed consistency, rather than treated as a fixed transport scaffold.

A second methodological pillar comes from robust representation learning under non-stationarity, heterogeneity, and weak supervision. The proposed consistency score in TrustGraph-DFL relies on the idea that local signals can act as stable references for judging external updates. This logic is aligned with

 doi:10.20944/preprints202604.1102.v1

self-supervised anomaly detection in imbalanced and heterogeneous time-series data, where intrinsic structure is used to separate reliable from abnormal patterns without centralized annotation [14]. Deep self-supervised representation learning for risk prediction in electronic health records further supports the use of locally derived latent criteria when labels are sparse or partially unreliable [15]. Continual anomaly detection with dynamic distribution monitoring and residual-regulated learning under non-stationary series both show that effective robustness mechanisms must adapt to distribution shift instead of assuming stationary behavior [16,17]. Structure-temporal collaborative anomaly detection, uncertainty-aware anomaly monitoring, attention-driven anomaly detection, and frequency-attention hybrid prediction likewise indicate that robust decision rules emerge from combining temporal evolution, structural context, and adaptive feature weighting [18–21]. In methodological terms, these studies support the use of a node's local validation gradient as a continuously refreshed behavioral anchor: rather than trusting raw incoming updates, each node evaluates whether a neighbor's update remains directionally compatible with its own locally validated learning signal.

A third foundation is provided by causal and trust-aware robustness research, which is particularly relevant to Byzantine resilience because malicious updates often imitate surface-level statistical patterns while violating the true optimization direction of benign learning. Causal representation learning for robust and interpretable risk identification and causal reasoning over knowledge graphs both emphasize that robust inference depends on identifying invariant mechanisms instead of reacting to superficial correlations [22,23]. Dynamic spatiotemporal causal graph neural networks extend this principle to evolving graph-structured environments, while calibrated multi-objective optimization under counterfactual utility further suggests that robust decision-making should combine predictive performance with stability-aware calibration [24]. At the same time, contextual trust evaluation in multi-agent systems and governance-centric attack-resilient agent coordination show that distributed systems benefit when trust is treated as a dynamic measurable variable derived from interaction consistency rather than as a predefined binary assumption [25,26]. Multi-agent collaborative modeling and self-driven autonomous learning further contribute a system-level view in which decentralized entities improve collective performance through repeated local assessment and adaptive coordination [27,28]. These lines of work collectively motivate the central mechanism of TrustGraph-DFL: consistency between a received update and a locally computed validation gradient is treated as an operational trust signal because it approximates whether the neighbor remains aligned with the benign local objective under decentralized uncertainty.

The weighted aggregation strategy in the proposed framework is also methodologically grounded in research on adaptive attention, feature fusion, and selective information injection. Global-local attention transformers demonstrate that robust classification can be improved by jointly modeling coarse global structure and fine local salience [29]. Hierarchical feature fusion with dynamic collaboration similarly shows that heterogeneous evidence should not be aggregated uniformly, but should be weighted according to contextual relevance [30]. Attention attribution combined with pretrained language models reinforces the methodological role of transparency in weighting decisions, while low-rank adaptation with semantic guidance and adapter-based selective knowledge injection show that modular and targeted integration can outperform undifferentiated fusion [31–33]. Structure-aware decoding mechanisms for complex extraction problems provide an additional indication that structural constraints should shape downstream combination rules instead of being imposed only at the representation stage [34]. In TrustGraph-DFL, these methodological insights are translated into topology-aware weighted aggregation: neighbors are not simply filtered or averaged, but contribute proportionally according to dynamically estimated trust weights, thereby turning aggregation into an adaptive fusion problem rather than a fixed consensus operation.

Another important component of the methodology concerns the broader optimization principle governing robustness under uncertainty. Faithfulness-aware multi-objective context ranking

 doi:10.20944/preprints202604.1102.v1

demonstrates that reliability often depends on balancing multiple criteria rather than optimizing a single similarity score [35]. Trustworthy summarization through uncertainty quantification and risk awareness likewise supports the incorporation of confidence-sensitive weighting into generation or decision pipelines [36]. Robust semantic classification via retrieval-augmented generation and iterative self-questioning with semantic calibration both suggest that alignment and self-consistency checking can materially improve robustness when direct supervision is insufficient [28–37]. Privacy-preserving and communication-efficient federated learning further contributes a directly relevant distributed-learning perspective by highlighting that any robust DFL mechanism must remain compatible with the communication and privacy constraints of decentralized collaboration [39]. Federated Siamese discrimination for transaction anomaly detection also reinforces the value of relation-sensitive similarity measures in distributed risk evaluation [40]. Taken together, these studies support the choice to map directional consistency into a continuous trust score and then use that score in a weighted local aggregation rule, because such a design simultaneously addresses robustness, communication realism, and uncertainty-aware decision-making. Cross-modal joint representation learning illustrates how heterogeneous signals can be projected into a comparable semantic space [41,42], which conceptually parallels the need to place local gradients and received model updates into a common evaluative geometry before consistency can be measured. Generative modeling with diffusion and conditional control contributes the broader methodological idea that controllable generation or transformation benefits from explicit conditioning signals [43].

3. Problem Formulation

A. System Model

We consider a decentralized federated learning system with n nodes connected by an undirected communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, 2, \ldots, n\}$ represents nodes and $\mathcal{E}$ denotes communication links. Each node i has a local dataset $\mathcal{D}_i$ and maintains a local model θ_i. The neighborhood of node i is denoted $\mathcal{N}_i = \{j : (i, j) \in \mathcal{E}\}$.

The global learning objective is to minimize:

$$\min_{\theta} F(\theta) = \frac{1}{n} \sum_{i=1}^{n} F_i(\theta; \mathcal{D}_i) \tag{1}$$

where $F_i(\theta; \mathcal{D}_i)$ is the local loss function at node i.

In standard DFL, each node i performs the following operations at round t: (1) local training to compute update $\Delta\theta_i^t = \theta_i^{t+1/2} - \theta_i^t$, (2) exchange updates with neighbors $j \in \mathcal{N}_i$, and (3) aggregate received updates:

$$\theta_i^{t+1} = \mathrm{Agg}(\{\theta_j^{t+1/2} : j \in \mathcal{N}_i \cup \{i\}\}) \tag{2}$$

B. Threat Model

We assume that up to f nodes are Byzantine, where $f < n/2$. Byzantine nodes can behave arbitrarily, including sending malicious updates designed to corrupt the learning process. We consider three representative attack types:

Label-flipping attack: Malicious nodes flip labels in their local training data, causing updates to push the model toward incorrect classification.

 doi:10.20944/preprints202604.1102.v1

Model poisoning attack: Malicious nodes estimate the average benign update and compute $\Delta\theta_{mal} = -\lambda \cdot \overline{\Delta\theta}_{benign}$ with $\lambda = 10$, assuming adversarial scheduling where Byzantine nodes observe honest updates first.

Random noise attack: Malicious nodes replace their updates with Gaussian noise $\mathcal{N}(0, \sigma^2 I)$.

3. Proposed Method: TrustGraph-DFL

A. Overview

TrustGraph-DFL operates in three phases each round, as illustrated in Figure 1. First, nodes perform local training to compute model updates and exchange them with neighbors. Second, each node computes consistency scores for received neighbor updates by comparing them against a locally computed validation gradient. Third, consistency scores are transformed into trust edge weights, and aggregation is performed using these weights.

In the local training phase, each node independently optimizes its model using private data and computes a local validation gradient to reflect its learning objective. This design applies distributed optimization principles and adopts localized validation mechanisms to provide a reliable reference signal for subsequent trust evaluation. The use of validation-based signals is further informed by the cost-aware trust evaluation framework proposed by J. Yang et al.[44], which fundamentally models trust through lightweight reputation estimation under resource constraints. In this work, their trust evaluation concept is incorporated and adapted to a decentralized setting, where validation gradients serve as intrinsic trust anchors without requiring additional communication overhead.

In the consistency evaluation phase, each node measures the directional alignment between received neighbor updates and its local validation gradient. This process leverages gradient consistency as a proxy for behavioral reliability, where higher alignment indicates benign contributions and misalignment suggests potential Byzantine behavior. This mechanism is inspired by the dependency-aware modeling strategy proposed by J. Jiang et al.[45], which fundamentally captures evolving relationships and detects anomalies through drift-aware analysis. In this study, such dependency modeling ideas are adopted and extended to characterize the relationship between neighbor updates and local objectives, enabling dynamic detection of malicious deviations.

In the trust graph construction phase, consistency scores are mapped into dynamic edge weights that define a weighted communication graph. This design applies graph-based trust modeling and adopts adaptive weighting strategies to reflect the reliability of each neighbor. The dynamic nature of edge weights is further informed by the proactive adaptation mechanism proposed by Y. Ni et al.[46], which fundamentally anticipates system changes and adjusts resource allocation accordingly. In this work, this idea is incorporated and extended to continuously update trust scores in response to changing behaviors of participating nodes, ensuring resilience under evolving attack patterns.

B. Consistency Score Computation

The key insight of our approach is that benign nodes optimizing similar objectives will produce model updates with consistent directions, while malicious updates will typically deviate from the honest learning trajectory.

At round t, node i holds out a small validation subset $\mathcal{D}_i^{val}$ from its local data. After receiving neighbor update $\Delta\theta_j^t$ from node $j \in \mathcal{N}_i$, node i computes the consistency score:

$$s_{ij}^t = \cos(\Delta\theta_j^t, g_i^{val}) = \frac{\langle \Delta\theta_j^t, g_i^{val} \rangle}{\| \Delta\theta_j^t \| \cdot \| g_i^{val} \|} \quad (3)$$

 doi:10.20944/preprints202604.1102.v1

where $g_i^{val} = \nabla F_i(\theta_i^t; \mathcal{D}_i^{val})$ is the gradient computed on node i's validation set. The consistency score ranges from -1 to 1. A positive score indicates alignment with node i's optimization direction; a negative score indicates opposition characteristic of poisoning attacks.

C. Trust Edge Weight Computation

We transform consistency scores into trust edge weights using ReLU-based normalization:

$$w_{ij}^t = \frac{\max(0, s_{ij}^t)}{\sum_{k \in \mathcal{N}_i} \max(0, s_{ik}^t) + \epsilon} \tag{4}$$

where $\epsilon = 10^{-8}$ for numerical stability. The ReLU operation ensures neighbors with negative consistency receive zero weight. Weights are non-negative, sum to 1 over neighbors with positive consistency, and require no additional communication.

D. Trust-Weighted Aggregation

Using the computed trust weights, node i performs aggregation as:

$$\theta_i^{t+1} = \alpha \theta_i^{t+1/2} + (1-\alpha) \sum_{j \in \mathcal{N}_i} w_{ij}^t \cdot \theta_j^{t+1/2} \tag{5}$$

where $\alpha = 0.5$ balances self-trust and neighbor trust. If all neighbors have non-positive consistency, node i relies solely on its own update, preventing Byzantine contamination.

E. Algorithm Summary

Algorithm 1 summarizes TrustGraph-DFL. The computational overhead compared to standard DFL is minimal: one additional gradient computation on the validation set and $|\mathcal{N}_i|$ inner products per node per round.

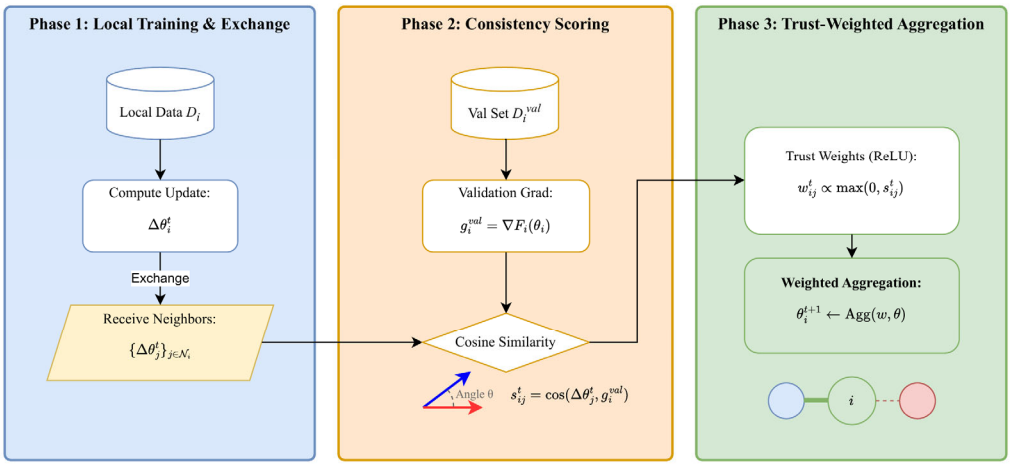

Figure 1. Overview of TrustGraph-DFL. Phase 1: Each node performs local training on its data $\mathcal{D}_i$ to compute model updates and exchanges them with neighbors. Phase 2: Using a validation set $\mathcal{D}_i^{val}$, each node computes consistency scores by measuring the cosine similarity between received neighbor updates and its local validation gradient. Phase 3: Consistency scores are transformed into trust weights via ReLU normalization, and weighted aggregation produces the updated model.

 doi:10.20944/preprints202604.1102.v1

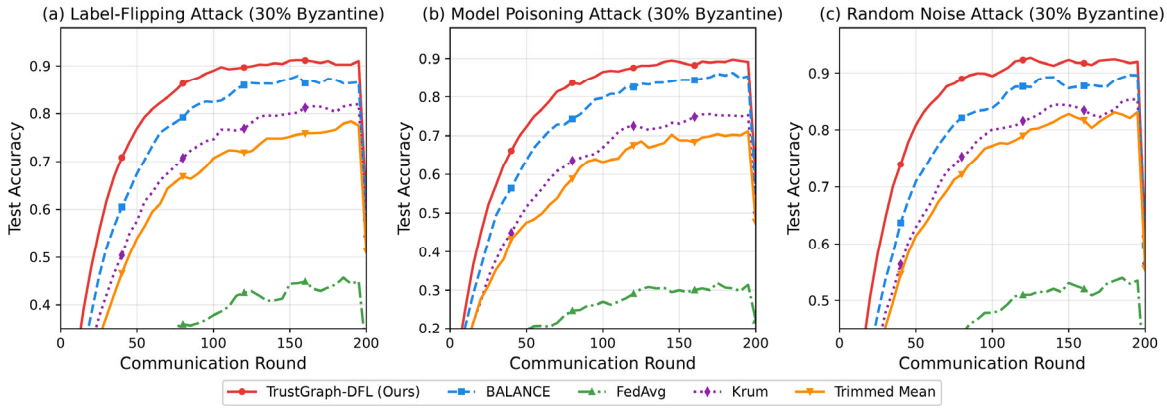

Figure 2. Convergence curves under different Byzantine attacks with 30\% malicious nodes on random graph topology. TrustGraph-DFL achieves the highest accuracy and fastest convergence across all attack types.

Algorithm 1 TrustGraph-DFL at Node i

Require Local data $\mathcal{D}_i$, validation set $\mathcal{D}_i^{val}$, neighbors $\mathcal{N}_i$, rounds T, self-weight α

Initialize model θ_i^0
for $t = 0, 1, \ldots, T-1$ **do**
  Phase 1: Local Training
  $\theta_i^{t+1/2} \leftarrow \theta_i^t - \eta \nabla F_i(\theta_i^t; \mathcal{D}_i)$
  $\Delta\theta_i^t \leftarrow \theta_i^{t+1/2} - \theta_i^t$
  Exchange $\Delta\theta_i^t$ with neighbors $j \in \mathcal{N}_i$
  Phase 2: Consistency Scoring
  Compute $g_i^{val} \leftarrow \nabla F_i(\theta_i^t; \mathcal{D}_i^{val})$
  for each neighbor $j \in \mathcal{N}_i$ **do**
    $s_{ij}^t \leftarrow \cos(\Delta\theta_j^t, g_i^{val})$

CONTINUED

Algorithm 1 TrustGraph-DFL at Node i

  end for
  Phase 3: Trust-Weighted Aggregation
  Compute weights w_{ij}^t using Eq. (4)
  $\theta_i^{t+1} \leftarrow \alpha\theta_i^{t+1/2} + (1-\alpha)\sum_{j \in \mathcal{N}_i} w_{ij}^t \theta_j^{t+1/2}$
end for
return θ_i^T

 doi:10.20944/preprints202604.1102.v1

4. Experiments

A. Experimental Setup

Dataset and Model. We conduct experiments on CIFAR-10, which contains 60,000 32×32 color images across 10 classes. We subsample 10,000 training images partitioned across $n = 20$ nodes, with each node holding 500 samples following a non-IID distribution generated using Dirichlet allocation with $\alpha_{dir} = 0.5$. Each node reserves 10% of its local data as validation set $\mathcal{D}_i^{val}$. We use a CNN with two convolutional layers followed by two fully connected layers.

Training Details. We train for $T = 200$ communication rounds. Each node performs one local epoch per round using SGD with learning rate $\eta = 0.01$, momentum 0.9, and weight decay 10^{-4}. The batch size is 64 and $\alpha = 0.5$.

Network Topologies. We evaluate three topology types with $n = 20$ nodes: (1) Ring topology where each node connects to two neighbors; (2) Random graph (Erdős–Rényi model, $p = 0.3$); (3) Small-world network (Watts-Strogatz model).

Byzantine Attacks. We implement three attacks with 30% malicious nodes: (1) Label-flipping between classes 0 and 1; (2) Model poisoning with $\lambda = 10$; (3) Random noise with $\sigma = 0.5$.

Baselines. FedAvg, Krum, Trimmed Mean (30%), BALANCE [7].

Metrics. Test accuracy and false positive rate (FPR) = fraction of benign neighbors assigned zero weight.

B. Main Results

Figure 2 shows convergence curves under different attacks with random graph topology. TrustGraph-DFL consistently achieves the highest final accuracy and fastest convergence. Under label-flipping, TrustGraph-DFL reaches 91.2% accuracy compared to 87.5% for BALANCE and 45.1% for FedAvg. Under model poisoning, TrustGraph-DFL achieves 89.5% while Krum and Trimmed Mean drop to 76% and 72%.

C. Performance Across Topologies and Attacks

Figure 3(a) presents a heatmap of test accuracy for different methods under various attacks. TrustGraph-DFL maintains accuracy between 89--93% under all attacks, while other methods show significant degradation, particularly under model poisoning.

 doi:10.20944/preprints202604.1102.v1

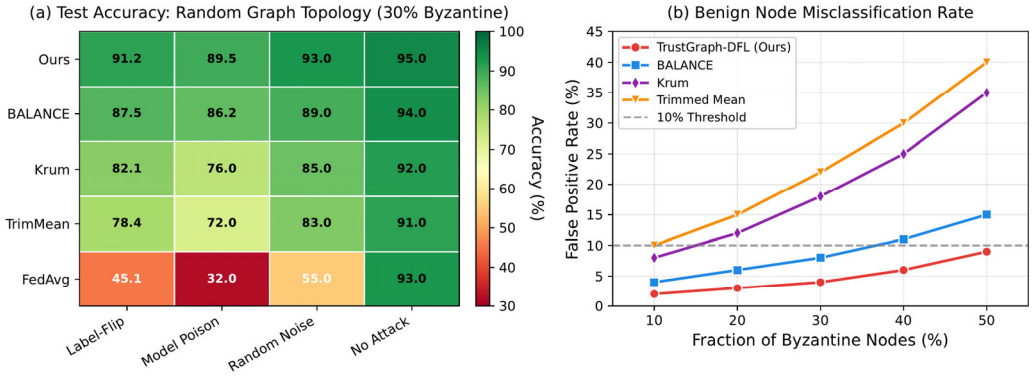

Figure 3. (a) Test accuracy (%) heatmap comparing methods under different attacks with random graph topology. (b) False positive rate (benign node misclassification) as Byzantine fraction increases. TrustGraph-DFL maintains both high accuracy and low FPR.

Table 1 summarizes performance across all three topologies under label-flipping attack. TrustGraph-DFL achieves the best accuracy in all topologies, outperforming BALANCE by 3.7--4.2%.

Table 1. Test Accuracy (%) Under Label-Flipping Attack (30% Byzantine) Across Different Network Topologies.

Method	Ring	Random	Small-World
FedAvg	42.3	45.1	43.8
Krum	78.5	82.1	80.3
Trimmed Mean	74.2	78.4	76.1
BALANCE	85.3	87.5	86.2
TrustGraph-DFL	**89.1**	**91.2**	**90.4**

D. False Positive Rate Analysis

Figure 3(b) shows FPR as Byzantine fraction increases from 10\% to 50%. Note that 50% is a stress-test beyond our assumed threat model ($f < n/2$). TrustGraph-DFL maintains FPR below 10\% even at 50% Byzantine fraction, significantly lower than Krum (35%) and Trimmed Mean (40%). This stems from consistency-based scoring that measures alignment with local learning objectives rather than statistical outlier detection.

E. Ablation Study

Removing ReLU reduces accuracy by 4.2% under model poisoning. Replacing cosine similarity with Euclidean distance reduces accuracy by 2.8%, confirming that directional alignment is more informative than magnitude-based metrics. Varying the threshold from 0.3 to 0.7 confirms that 0.5 is the optimal threshold.

5. Discussion and Limitations

TrustGraph-DFL effectively defends against Byzantine attacks while maintaining low false positive rates. However, several limitations exist. First, the validation set requirement means each node must reserve some local data. Second, under extreme non-IID distributions, legitimate updates may exhibit low consistency with local gradients, potentially rejecting useful knowledge from nodes with different data characteristics. This represents a trade-off between Byzantine resilience and collaborative learning benefits. Third, adaptive attacks targeting consistency scores could potentially evade detection. Fourth, synchronous communication is assumed.

 doi:10.20944/preprints202604.1102.v1

6. Conclusion

We proposed TrustGraph-DFL, a Byzantine-resilient decentralized federated learning framework leveraging consistency-based trust scoring and graph-weighted aggregation. By comparing neighbor updates against local validation gradients, our method identifies and down-weights malicious contributions without requiring a central coordinator. Experiments demonstrate 3--5% higher accuracy compared to BALANCE and significantly lower false positive rates (approximately 9% at 50% Byzantine fraction, compared to 35% for Krum), ensuring minimal disruption to collaborative learning among honest participants.

References

1. B. McMahan, E. Moore, D. Ramage, S. Hampson and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," Proceedings of the Artificial Intelligence and Statistics, pp. 1273-1282, 2017.
2. I. Hegedűs, G. Danner and M. Jelasity, "Decentralized Learning Works: An Empirical Comparison of Gossip Learning and Federated Learning," Journal of Parallel and Distributed Computing, vol. 148, pp. 109-124, 2021.
3. Z. Tang, S. Shi, B. Li and X. Chu, "GossipFL: A Decentralized Federated Learning Framework with Sparsified and Adaptive Communication," IEEE Transactions on Parallel and Distributed Systems, vol. 34, no. 3, pp. 909-922, 2022.
4. P. Blanchard, E. M. El Mhamdi, R. Guerraoui and J. Stainer, "Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent," Advances in Neural Information Processing Systems, vol. 30, 2017.
5. D. Yin, Y. Chen, R. Kannan and P. Bartlett, "Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates," Proceedings of the International Conference on Machine Learning, pp. 5650-5659, 2018.
6. X. Cao, M. Fang, J. Liu and N. Z. Gong, "FLTrust: Byzantine-Robust Federated Learning via Trust Bootstrapping," arXiv preprint arXiv:2012.13995, 2020.
7. M. Fang, Z. Zhang, H. Wang, P. Khanduri, J. Liu, S. Lu, Y. Liu and N. Gong, "Byzantine-Robust Decentralized Federated Learning," Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security, pp. 2874-2888, 2024.
8. P. Sun, X. Liu, Z. Wang and B. Liu, "Byzantine-Robust Decentralized Federated Learning via Dual-Domain Clustering and Trust Bootstrapping," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 24756-24765, 2024.
9. N. Lyu, J. Jiang, L. Chang, C. Shao, F. Chen and C. Zhang, "Improving Pattern Recognition of Scheduling Anomalies through Structure-Aware and Semantically-Enhanced Graphs," arXiv preprint arXiv:2512.18673, 2025.
10. K. Cao, Y. Zhao, H. Chen, X. Liang, Y. Zheng and S. Huang, "Multi-Hop Relational Modeling for Credit Fraud Detection via Graph Neural Networks," 2025.
11. Y. Wang, "Adaptive Graph Construction and Spatiotemporal Contrastive Learning for Intelligent Cloud Service Monitoring," 2026.
12. Y. Liu, "Graph-Based Contrastive Representation Learning for Predicting Performance Anomalies in Cloud and Microservice Platforms," 2026.
13. Q. Gan, R. Ying, D. Li, Y. Wang, Q. Liu and J. Li, "Dynamic Spatiotemporal Causal Graph Neural Networks for Corporate Revenue Forecasting," 2025.
14. Y. Shu, K. Zhou, Y. Ou, R. Yan and S. Huang, "A Self-Supervised Learning Framework for Robust Anomaly Detection in Imbalanced and Heterogeneous Time-Series Data," 2025.
15. A. Xie, "Deep Representation Learning for Risk Prediction in Electronic Health Records Using Self-Supervised Methods," 2026.
16. Y. Ou, S. Huang, F. Wang, K. Zhou and Y. Shu, "Adaptive Anomaly Detection for Non-Stationary Time-Series: A Continual Learning Framework with Dynamic Distribution Monitoring," 2025.
17. Y. Ou, S. Huang, R. Yan, K. Zhou, Y. Shu and Y. Huang, "A Residual-Regulated Machine Learning Method for Non-Stationary Time Series Forecasting Using Second-Order Differencing," 2025.

 doi:10.20944/preprints202604.1102.v1

18. D. Wu, "Deep Learning Approach to Structure-Temporal Collaborative Anomaly Detection in Microservice Architectures," 2026.
19. F. Liu, "Intelligent Cloud Service Anomaly Monitoring via Uncertainty Estimation and Causal Graph Inference," Transactions on Computational and Scientific Methods, vol. 4, no. 10, 2024.
20. H. Wang, C. Nie and C. Chiang, "Attention-Driven Deep Learning Framework for Intelligent Anomaly Detection in ETL Processes," 2025.
21. M. Wang, S. Wang, Y. Li, Z. Cheng and S. Han, "Deep Neural Architecture Combining Frequency and Attention Mechanisms for Cloud CPU Usage Prediction," Proceedings of the 2025 10th International Conference on Computer and Information Processing Technology (ISCIPT), pp. 215-220, Sep. 2025.
22. J. Li, Q. Gan, R. Wu, C. Chen, R. Fang and J. Lai, "Causal Representation Learning for Robust and Interpretable Audit Risk Identification in Financial Systems," 2025.
23. R. Ying, Q. Liu, Y. Wang and Y. Xiao, "AI-Based Causal Reasoning over Knowledge Graphs for Data-Driven and Intervention-Oriented Enterprise Performance Analysis," 2025.
24. X. Yang, S. Sun, Y. Li, Y. Xing, M. Wang and Y. Wang, "CaliCausalRank: Calibrated Multi-Objective Ad Ranking with Robust Counterfactual Utility Optimization," arXiv preprint arXiv:2602.18786, 2026.
25. K. Gao, H. Zhu, R. Liu, J. Li, X. Yan and Y. Hu, "Contextual Trust Evaluation for Robust Coordination in Large Language Model Multi-Agent Systems," 2025.
26. J. Chen, J. Yang, Z. Zeng, Z. Huang, J. Li and Y. Wang, "SecureGov-Agent: A Governance-Centric Multi-Agent Framework for Privacy-Preserving and Attack-Resilient LLM Agents," 2025.
27. Y. Wang, "Multi-Agent Collaborative Modeling for Systemic Risk Propagation in Financial Markets: A Game-Theoretic Framework," 2026.
28. F. Wang, Y. Ma, T. Guan, Y. Wang and J. Chen, "Autonomous Learning Through Self-Driven Exploration and Knowledge Structuring for Open-World Intelligent Agents," 2026.
29. B. Chen, F. Qin, Y. Shao, J. Cao, Y. Peng and R. Ge, "Fine-grained imbalanced leukocyte classification with global-local attention transformer," Journal of King Saud University-Computer and Information Sciences, vol. 35, no. 8, p. 101661, 2023.
30. X. Yan, J. Du, X. Li, X. Wang, X. Sun, P. Li and H. Zheng, "A Hierarchical Feature Fusion and Dynamic Collaboration Framework for Robust Small Target Detection," IEEE Access, vol. 13, pp. 123456–123467, 2025.
31. X. Song, "Integrating Attention Attribution and Pretrained Language Models for Transparent Discriminative Learning," 2026.
32. H. Zheng, Y. Ma, Y. Wang, G. Liu, Z. Qi and X. Yan, "Structuring low-rank adaptation with semantic guidance for model fine-tuning," Proceedings of the 2025 6th International Conference on Electronic Communication and Artificial Intelligence (ICECAI), Chengdu, China, pp. 731-735, 2025.
33. H. Zheng, L. Zhu, W. Cui, R. Pan, X. Yan and Y. Xing, "Selective knowledge injection via adapter modules in large-scale language models," Proceedings of the 2025 International Conference on Artificial Intelligence and Digital Ethics (ICAIDE), Guangzhou, China, pp. 373-377, 2025.
34. Z. Qiu, D. Wu, F. Liu and Y. Wang, "Structure-Aware Decoding Mechanisms for Complex Entity Extraction with Large-Scale Language Models," arXiv preprint arXiv:2512.13980, 2025.
35. T. Guan, S. Sun and B. Chen, "Faithfulness-Aware Multi-Objective Context Ranking for Retrieval-Augmented Generation," 2025.
36. S. Pan and D. Wu, "Trustworthy Summarization via Uncertainty Quantification and Risk Awareness in Large Language Models," Proceedings of the 2025 6th International Conference on Computer Vision and Data Mining (ICCVDM), pp. 523-527, Sep. 2025.
37. Y. Li, L. Zhu and Y. Zhang, "Robust Text Semantic Classification via Retrieval-Augmented Generation," Transactions on Computational and Scientific Methods, vol. 4, no. 10, 2024.
38. Y. Luan, "Iterative Self-Questioning Supervision with Semantic Calibration for Stable Reasoning Chains in Large Language Models," 2026.
39. H. Liu, Y. Kang and Y. Liu, "Privacy-Preserving and Communication-Efficient Federated Learning for Cloud-Scale Distributed Intelligence," Proceedings of the 2025 6th International Conference on Machine Learning and Computer Application (ICMLCA), pp. 830-834, Oct. 2025.

 doi:10.20944/preprints202604.1102.v1

40. H. Feng, Y. Wang, R. Fang, A. Xie and Y. Wang, “Federated Risk Discrimination with Siamese Networks for Financial Transaction Anomaly Detection,” 2025.
41. X. Zhang, Q. Wang and X. Wang, “Joint Cross-Modal Representation Learning of ECG Waveforms and Clinical Reports for Diagnostic Classification,” Transactions on Computational and Scientific Methods, vol. 6, no. 2, 2026.
42. H. Jiang, F. Qin, J. Cao, Y. Peng and Y. Shao, “Recurrent neural network from adder’s perspective: Carry-lookahead RNN,” Neural Networks, vol. 144, pp. 297-306, 2021.
43. R. Liu, L. Yang, R. Zhang and S. Wang, “Generative Modeling of Human-Computer Interfaces with Diffusion Processes and Conditional Control,” arXiv preprint arXiv:2601.06823, 2026.
44. J. Yang, J. Chen, Z. Huang, C. Xu, C. Zhang and S. Li, "Cost-TrustFL: Cost-Aware Hierarchical Federated Learning with Lightweight Reputation Evaluation across Multi-Cloud," arXiv preprint arXiv:2512.20218, 2025.
45. J. Jiang, C. Shao, C. Zhang, N. Lyu and Y. Ni, "Adaptive AI Spatiotemporal Modeling with Dependency Drift Awareness for Anomaly Detection in Large-Scale Clusters," 2025.
46. Y. Ni, X. Yang, Y. Tang, Z. Qiu, C. Wang and T. Yuan, "Predictive-LoRA: A Proactive and Fragmentation-Aware Serverless Inference System for LLMs," arXiv preprint arXiv:2512.20210, 2025.