Distributed Federated Learning for Vehicular Network Security: Anomaly Detection Benefits and Multi-Domain Attack Threats

Utku Demir
Nexcepta Inc.
Gaithersburg, MD, USA

Yalin E. Sagduyu
Nexcepta Inc.
Gaithersburg, MD, USA

Tugba Erpek
Nexcepta Inc.
Gaithersburg, MD, USA

Hossein Jafari
Nexcepta Inc.
Gaithersburg, MD, USA

Sastry Kompella
Nexcepta Inc.
Gaithersburg, MD, USA

Mengran Xue
RTX BBN Technologies
Cambridge, MA, USA

ABSTRACT

In connected and autonomous vehicles, machine learning for safety message classification has become critical for detecting malicious or anomalous behavior. However, conventional approaches that rely on centralized data collection or purely local training face limitations due to the large scale, high mobility, and heterogeneous data distributions inherent in inter-vehicle networks. To overcome these challenges, this paper explores Distributed Federated Learning (DFL), whereby vehicles collaboratively train deep learning models by exchanging model updates among one-hop neighbors and propagating models over multiple hops. Using the Vehicular Reference Misbehavior (VeReMi) Extension Dataset, we show that DFL can significantly improve classification accuracy across all vehicles compared to learning strictly with local data. Notably, vehicles with low individual accuracy see substantial accuracy gains through DFL, illustrating the benefit of knowledge sharing across the network. We further show that local training data size and time-varying network connectivity correlate strongly with the model's overall accuracy. We investigate DFL's resilience and vulnerabilities under attacks in multiple domains, namely wireless jamming and training data poisoning attacks. Our results reveal important insights into the vulnerabilities of DFL when confronted with multi-domain attacks, underlining the need for more robust strategies to secure DFL in vehicular networks.

KEYWORDS

Distributed federated learning, vehicular networks, anomaly detection, deep learning, adversarial machine learning, security.

1 INTRODUCTION

The rapid evolution of connected and autonomous vehicles (CAVs) is reshaping modern transportation systems. Recent years have witnessed significant advancements in inter-vehicle communication, spurred by the emergence of connected and autonomous vehicles and the increasing demand for real-time data exchange. These developments, facilitated by protocols such as Dedicated Short-Range Communications (DSRC) and Cellular Vehicle-to-Everything (C-V2X), have paved the way for various applications in cooperative safety, traffic optimization, and entertainment services [5]. As vehicles increasingly rely on real-time data sharing and cooperative decision-making, ensuring the reliability and security of safety message exchange has become paramount. Anomaly detection, namely identifying malicious messages received by individual vehicles, is critical to preventing cascading failures and ensuring road safety. However, traditional anomaly detection approaches, which rely on centralized data aggregation or isolated local training, struggle to address the complexities of dynamic vehicular environments and remain insufficient in the face of highly mobile network topologies, geographically dispersed data, and stringent latency constraints.

Federated learning (FL) has emerged in vehicular networks as a means of leveraging the abundant distributed data generated by onboard sensors (e.g., LiDAR, radar, cameras) and network data (e.g., beacon messages). By training a shared model over multiple decentralized nodes, FL reduces the need for raw data transfer to a central server and offers privacy benefits, as only local model updates are communicated. FL finds various applications in vehicular networks. For security purposes, FL was applied for misbehavior detection of safety messages [3], anomaly detection in the Internet of Vehicles [12], vehicle trajectory prediction against cyber attacks [13] and V2X misbehavior detection in 5G edge networks [14].

Despite its benefits, most existing FL implementations still rely on a central server that aggregates and redistributes model parameters. Although effective in stable or low-mobility scenarios, this architecture encounters practical challenges in vehicular networks, where connectivity is transient and fixed infrastructure may be unavailable.

These limitations have prompted us to explore Distributed Federated Learning (DFL), which eliminates the reliance on a single central aggregator in favor of multiple local aggregations among neighboring vehicles [10]. DFL leverages the computational resources of individual vehicles while preserving data privacy, and enables vehicles to propagate updates through a multi-hop network, overcoming limitations associated with centralized systems by allowing faster adaptation and reducing single-point-of-failure risks. In DFL, vehicles engage in direct, one-hop model exchanges that propagate aggregated knowledge over multiple hops enhancing scalability and improving resilience against connectivity disruptions. Different modes of learning are illustrated in Figure 1.

DFL in vehicular networks faces several challenges. The dynamic topology and variable connectivity of inter-vehicle networks make consistent and timely model aggregation a complex task. Synchronization issues can emerge when nodes differ in connectivity or local data volume, potentially hampering the convergence and stability of the learning process. The diverse quality and quantity of local training data mean that some nodes may initially exhibit poor classification performance. By sharing model updates, however, DFL has the potential to lift these underperforming nodes, leading to an overall improvement in network-wide anomaly detection.

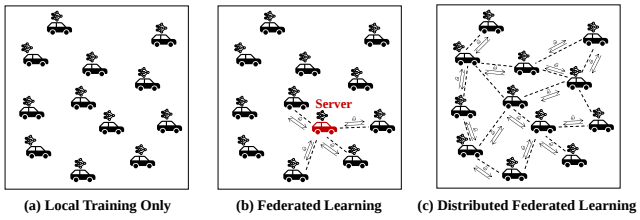


Figure 1: Different modes of learning: (a) local training only, (b) federated learning, (c) distributed federated learning.

In this paper, we leverage the Vehicular Reference Misbehavior (VeReMi) Extension Dataset [4] to investigate the performance and security of DFL for safety message classification. We show that DFL can significantly improve classification accuracy compared to the local learning approach that relies solely on individual vehicle data. Vehicles that initially exhibit low individual accuracy due to limited local training samples experience gains when participating in DFL. We also uncover a strong correlation of overall model accuracy with local training data size and network connectivity over time.

Another critical consideration in adopting DFL is its vulnerability to multi-domain attacks. The decentralized nature of DFL, while reducing dependency on central infrastructure, also exposes the system to new security threats. In the wireless domain, jamming attacks may disrupt communication channels used for model propagation [8, 9, 11]. Concurrently, adversaries may launch data poisoning attacks, wherein compromised vehicles inject misleading information into the training process [1, 6, 7]. DFL in a vehicular network setting is not immune to these attacks. Jamming attacks may prevent the exchange of model updates between vehicles, limiting the network coverage for DFL. Data poisoning attacks may contaminate local datasets to mislead the model, propagating misinformation and eventually degrading global performance. We show that jamming and poisoning attacks can be either individually or jointly launched to reduce the DFL accuracy effectively. Our results uncover the characteristics of multi-domain threats to DFL and provide insights on vulnerabilities of DFL in practice. As vehicular networks become more interconnected, ensuring the robustness of DFL against multi-domain threats is of utmost importance. This paper makes the following contributions:

(1) *DFL Framework Design*: We apply DFL to anomaly detection in inter-vehicle networks, enabling one-hop model exchanges and multi-hop propagation without central server.
(2) *Performance Analysis*: Through extensive experiments with the VeReMi Extension Dataset, we demonstrate that DFL significantly outperforms local learning in terms of classification accuracy, benefiting nodes with limited data.
(3) *Vulnerability Characterization*: We perform a detailed analysis of DFL under multi-domain attacks, including jamming and poisoning attacks, highlighting vulnerabilities of DFL to individual and joint attack effects.

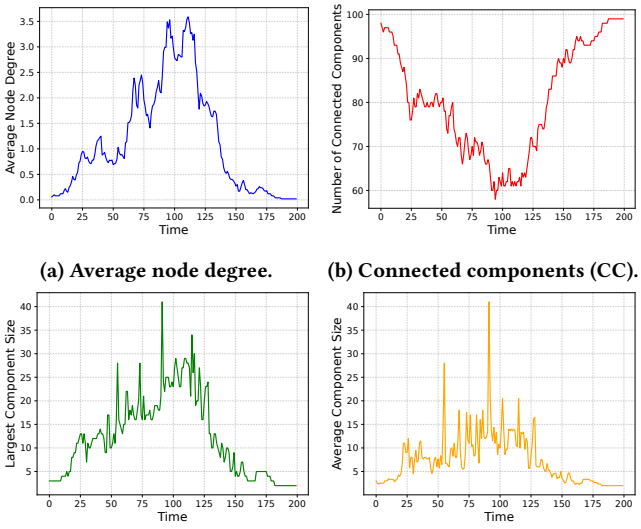
The remainder of the paper is organized as follows. Section 2 introduces the system model for DFL in inter-vehicle networks. Section 3 evaluates DFL performance for anomaly detection. Section 4 presents attacks on DFL. Section 5 concludes the paper.

2 SYSTEM MODEL

2.1 Vehicular Network Dataset

The VeReMi Extension dataset was developed for malicious behavior detection in Cooperative Intelligent Transport Systems (C-ITS) [4]. Each vehicle sends basic safety messages (BSMs) to the other vehicles in its communication range during its operation. This dataset is a collection of these BSMs and introduces both malfunctions (non-malicious errors due to faulty sensors) and attacks (intentional disruptions). Each message is tagged with type of message, timestamp of transmission and reception, sender identity, position, speed, acceleration, and heading (with and without noise).

In this paper, we use the morning peak traffic dataset and focus on the Denial of Service (DoS) attacks in terms of high-frequency message flooding. There are a total of 64, 779 BSMs across all the users. We separate the data for training, validation and testing. We train a deep learning model at each vehicle using 22 features [2]. There are 100 vehicles (nodes) in the vehicular network, 94 of them with (training and test) data and 6 of them without data. The connectivity of nodes is determined by the adjacency matrix that changes over time with node mobility. Network metrics over time (average node degree, number of connected components, average connected component size, and largest connected component size) are shown in Figure 2. The heatmap for connectivity among nodes aggregated over time is shown in Figure 3. We evaluate the relationship between the DFL performance and network properties in Section 3.



(a) Average node degree. **(b) Connected components (CC).**

(c) Largest CC size. **(d) Average CC size.**

Figure 2: Network metrics over time.

2.2 Distributed Federated Learning Operation

In FL, each vehicle trains a local model and shares only model updates instead of raw data. However, standard FL relies on a central server for model aggregation, which introduces bottlenecks and single points of failure. To address these challenges, DFL supports vehicles to exchange model updates directly with their one-hop

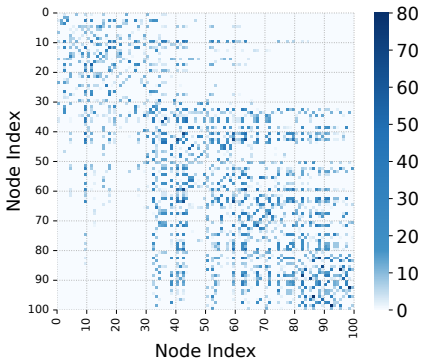


Figure 3: Aggregated adjacency matrix (link persistence).

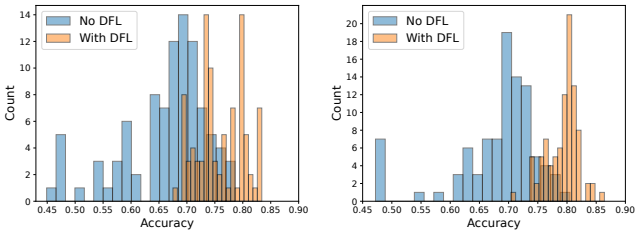
neighbors in a peer-to-peer manner. Instead of requiring a central aggregator, models are aggregated locally and propagated through the network in a multi-hop fashion, enabling decentralized knowledge sharing while maintaining privacy and reducing reliance on fixed infrastructure. With DFL, each vehicle acts as an autonomous node that collects sensor data, processes it locally, and participates in collaborative learning. Main components of DFL training are:

(1) *Initial Local Training*: Each vehicle maintains a local dataset gathered from onboard sensors and communication modules (e.g., beacon messages, BSMs). Vehicles train deep learning models independently based on their local data.
(2) *Model Exchange with One-Hop Neighbors*: Instead of sending data to a central server, each vehicle shares its locally trained models with one-hop neighbor vehicles.
(3) *Local Aggregation and Multi-Hop Propagation*: Each vehicle aggregates received model updates from its neighbors to its local model through federated averaging. The updated model is propagated in a multi-hop manner across the network.
(4) *Iterative Learning*: Vehicles continue training and exchanging models over multiple rounds. Over time, this process allows all vehicles to collaboratively learn a global representation of the model without central coordination.

3 ANOMALY DETECTION WITH DISTRIBUTED FEDERATED LEARNING

In DFL, nodes collectively train their models to classify incoming messages as benign or malicious. As a benchmark, we consider local training only without model exchange. For comparison purposes, we consider two types of DNN models, one small model and one large model, with architectures shown in Table 1. The nodes have training data sets ranging from 86 to 2514 samples, and the average size of the training data per node is 689 samples. On the other hand, nodes have test (inference) datasets ranging from 3 to 114 samples, and the average test data size per node is 36 samples.

We measure different accuracy metrics, including average, minimum, and maximum accuracy and its standard deviation across all nodes. Accuracy results are shown in Table 2. The corresponding histograms of accuracy values achieved by all nodes are shown in Figure 4. In both models, DFL improves every node's classification



(a) When the small DNN is used. **(b) When the large DNN is used.**

Figure 4: Histogram of accuracies with and without DFL.

accuracy, significantly raises the average and minimum accuracies, and reduces the standard deviation compared to local training alone. Overall, the large DNN model achieves higher accuracy compared to the small DNN model with DFL. Thus, for the rest of the paper, we continue with the large DNN model.

Table 1: DNN architectures.

	Small DNN		Large DNN	
Layer	Size	Activation	Size	Activation
Input	22	–	22	–
Hidden 1	16	ReLU	128	ReLU
Hidden 2	8	ReLU	32	ReLU
Output	2	Softmax	2	Softmax

Table 2: Performance with and without DFL.

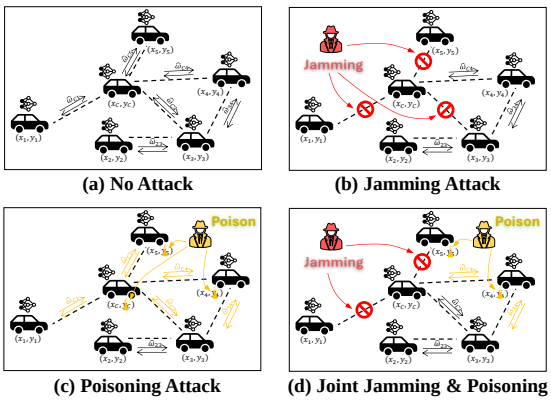
(a) Small DNN.

Metric	No DFL	With DFL	DFL Improvement
Average Accuracy	0.6625	0.7592	14.60%
Minimum Accuracy	0.4490	0.6750	50.33%
Maximum Accuracy	0.7860	0.8339	6.09%
Standard Deviation	0.0762	0.0426	44.09%

(b) Large DNN.

Metric	No DFL	With DFL	DFL Improvement
Average Accuracy	0.6811	0.8004	17.52%
Minimum Accuracy	0.4720	0.7040	49.15%
Maximum Accuracy	0.7870	0.8640	9.78%
Standard Deviation	0.0732	0.0275	62.43%

Next, we analyze the correlation of the DFL accuracy with the local (individual) learning accuracy, the training data size, and the network properties including the node degree and the average size of connected components over time. We make the following definitions. $\mathcal{N}$ is the set of nodes with (training and test) data. $\boldsymbol{m}$ is the vector of training data sizes, where $\boldsymbol{m}_i$ is the training data size of node $i \in \mathcal{N}$. $\boldsymbol{a}_{DFL}$ is the vector of DFL accuracies, where $\boldsymbol{a}_{DFL,i}$ is the DFL accuracy of node $i \in \mathcal{N}$. $\boldsymbol{a}_{LL}$ is the vector of local learning accuracies without DFL, where $\boldsymbol{a}_{LL,i}$ is the local learning accuracy of node $i \in \mathcal{N}$. $\boldsymbol{d}$ is the average incoming node degree, where $\boldsymbol{d}_i$ is the average number of incoming connections to that node $i \in \mathcal{N}$



(a) No Attack (b) Jamming Attack (c) Poisoning Attack (d) Joint Jamming & Poisoning

Figure 5: Jamming and poisoning attacks on DFL.

over time. C is the average size of connected components for nodes, where C_i is the average size of connected components that node $i \in \mathcal{N}$ belongs to over time. $\boldsymbol{c}$ is the ratio of connected times for nodes, where c_i is the ratio of times when node $i \in \mathcal{N}$ is connected to at least one other node over time.

Table 3: Correlation analysis.

Pearson Correlation		Spearman's Rank Correlation	
Terms	Coefficient	Terms	Coefficient
$\rho(\boldsymbol{a}_{DFL}, \boldsymbol{a}_{LL})$	0.3386	$s(\boldsymbol{a}_{DFL}, \boldsymbol{a}_{LL})$	0.2895
$\rho(\boldsymbol{a}_{DFL}, \boldsymbol{m})$	0.4365	$s(\boldsymbol{a}_{DFL}, \boldsymbol{m})$	0.4548
$\rho(\boldsymbol{a}_{LL}, \boldsymbol{m})$	0.3074	$s(\boldsymbol{a}_{LL}, \boldsymbol{m})$	0.3561
$\rho(\boldsymbol{a}_{DFL}, \boldsymbol{d})$	0.4552	$s(\boldsymbol{a}_{DFL}, \boldsymbol{d})$	0.4988
$\rho(\boldsymbol{a}_{DFL}, \boldsymbol{C})$	0.5771	$s(\boldsymbol{a}_{DFL}, \boldsymbol{C})$	0.6385
$\rho(\boldsymbol{a}_{DFL}, \boldsymbol{c})$	0.4251	$s(\boldsymbol{a}_{DFL}, \boldsymbol{c})$	0.4808

Let $\rho(\boldsymbol{x}, \boldsymbol{y})$ and $s(\boldsymbol{x}, \boldsymbol{y})$ denote the Pearson correlation coefficient and Spearman's rank correlation coefficient, respectively, between $\boldsymbol{x}$ and $\boldsymbol{y}$. For the large DNN, the correlation analysis results are given in Table 3 for Pearson correlation coefficient and Spearman's rank correlation. DFL accuracy and local learning accuracy are positively correlated with each other, and DFL accuracy has higher correlation with training data size compared to local learning accuracy. While local learning accuracy is not correlated with network properties in general, DFL accuracy is highly correlated with node degree, average size of connected components, and ratio of connected time. We utilize this correlation in Section 4 to define attacks on DFL.

4 MULTI-DOMAIN ATTACKS ON DISTRIBUTED FEDERATED LEARNING

4.1 Jamming Attacks

Jamming attacks pose a significant threat to the performance and reliability of DFL in inter-vehicle networks. DFL relies on timely and repeated communication among neighbors to exchange and aggregate local model updates. Jamming can severely impair the propagation of learning updates (from Figure 5a to Figure 5b) and degrade the overall model quality. Nodes isolated by jamming cannot contribute their local knowledge to the network. Their neighbors miss out on potentially valuable updates, leading to incomplete or biased aggregation. Nodes affected by jamming can only continue local training, which degrades performance without DFL.

We assess DFL's robustness to jamming attacks of varying intensity by selectively isolating a subset of nodes, i.e., severing their incoming connections, and measuring the resulting performance. In practice, jamming can be performed with directional transmissions without affecting the BSMs received at other vehicles. First, we select this group of nodes as the $TopK_J$ performers in the DFL learning scenario, where J stands for jamming. The reason for this type of selection is that DFL allows users with multiple neighbors to exchange information (model weights) and learn more effectively. This also provides a higher attack success rate compared to random jamming. Later, we switch to selecting the set of jammed nodes based on the network connectivity properties. By gradually cutting the connections of the top performers in DFL, we eventually reduce the network down to local learning (no FL), which allows us to show how DFL enhances learning within the network.

We provide the results for average and minimum accuracies in Table 4. We start from DFL with no attacks and increase the number of users to jam from 10 to all users, which comes down to local learning, by effectively cutting all incoming connections to users. DFL achieves an average accuracy of 0.8004 among all users within the network, where the minimum accuracy is 0.7034. It is evident that as we cut the top performing users in DFL from the network and they can no longer provide their models to other participating nodes, the accuracy across the network decreases. When all the connections are cut, DFL is reduced to local learning with the average accuracy of 0.6811 and minimum accuracy of 0.4720, underlining the impact of jamming on DFL performance.

Table 4: Effects of jamming attacks on DFL performance.

Jammed Nodes	Average Accuracy	Minimum Accuracy	Jammed Nodes	Average Accuracy	Minimum Accuracy
None	0.8004	0.7034	$Top60_J$	0.7266	0.5199
$Top10_J$	0.7976	0.6949	$Top70_J$	0.7187	0.4749
$Top20_J$	0.7763	0.6909	$Top80_J$	0.7036	0.4729
$Top30_J$	0.7690	0.6869	$Top90_J$	0.6911	0.4729
$Top40_J$	0.7680	0.6800	All	0.6811	0.4720
$Top50_J$	0.7412	0.5600	–	–	–

4.2 Poisoning Attacks

A training data poisoning attack is an adversarial machine learning attack that manipulates the training dataset by injecting misleading or malicious data samples to degrade the performance of a machine learning model. One common strategy is label manipulation, where the attacker intentionally assigns incorrect labels to training data samples (Figure 5c). Similar to the jamming case, we evaluate the performance of the DFL model using different poisoning intensities.

We assume that the training data labels are flipped for a certain set of nodes with probability p_a. We select $TopK_P$ nodes to poison, where P stands for poisoning. Some nodes cannot achieve high classification accuracy with only local learning. Thus, when their labels are flipped after poisoning, they may deceptively improve

their classification accuracy. Some nodes already have high classification accuracy with local learning. Poisoning the training data samples of those nodes effectively reduces the overall classification accuracy. Contrary to the jamming case, we determine the set of users to poison from the best performers in the local learning (no DFL) case before the attack. While this process leads to an effective poisoning attack, it may not be possible for an attacker to know in advance which nodes to poison. Later, we replace this selection of poisoned nodes based on network connectivity properties that may be observed by the attacker over the air. For $TopK_P$, we select 47 as the half of the total number of nodes with data, and 25 and 70 as the intermediate node numbers. We select $p_a = \{0.25, 0.5, 0.75, 1\}$ as the probability of poisoning the labels of training data samples (from the $TopK_P$ nodes to be poisoned).

We present poisoning attack results in Table 5 and compare them with the no-attack case. As we increase the poisoning attack intensity (higher K in $TopK_P$, with higher probability of label flipping, p_a) average accuracies drop, eventually settling at basically random guessing, i.e., 0.5314 when $p_a = 1$ and top 70 high accuracy nodes, $Top70_P$, are poisoned. Similarly, minimum accuracy values drop significantly, even below a random guessing case, e.g., 0.3149 when $p_a = 1$ and top 70 best accuracy nodes, $Top70_P$, are poisoned. We observe that label poisoning attacks do not remain localized – they propagate over multiple hops, gradually corrupting the learning process and leading to classification errors.

Table 5: Effects of poisoning attacks on DFL performance.

(a) Average accuracy.

Poisoned Nodes	Poisoning Probability			
	$p_a = 0.25$	$p_a = 0.5$	$p_a = 0.75$	$p_a = 1$
None	0.8004			
$Top25_P$	0.7685	0.7532	0.7435	0.7378
$Top47_P$	0.7577	0.7313	0.7309	0.6619
$Top70_P$	0.7540	0.7166	0.6227	0.5314

(b) Minimum accuracy.

Poisoned Nodes	Poisoning Probability			
	$p_a = 0.25$	$p_a = 0.5$	$p_a = 0.75$	$p_a = 1$
None	0.7040			
$Top25_P$	0.6869	0.6779	0.6420	0.6480
$Top47_P$	0.6869	0.6269	0.5320	0.4189
$Top70_P$	0.6869	0.5939	0.4680	0.3149

4.3 Joint Jamming and Poisoning Attacks

Next, we evaluate the effect of the simultaneous jamming and poisoning attacks (Figure 5d) across a wide range of attack intensities. We use $p_a = \{0.5, 1\}$, $TopK_J = \{25, 47, 70\}$, and $TopK_P = \{25, 47, 70\}$ in the simulations. Results are provided in Table 6. The average accuracy drops within each subtable as the jamming intensity increases, i.e., higher $TopK_J$, and all of these accuracies are lower than the no attack DFL accuracy of 80%. Within each subtable higher p_a results in lower accuracy, as expected.

By comparing Table 6a and Table 6c, we see that when more $TopK_P$ nodes are poisoned, overall accuracy consistently decreases and the accuracies in Table 6c are consistently lower than their counterparts in Table 6a. However, when we compare Table 6c with Table 6e, as $TopK_J$ nodes are jammed, the resulting accuracies are higher compared to their counterparts in Table 6c. The reason is that as more nodes are included in $TopK_P$, it is more likely that the nodes with low accuracy will be poisoned, as well. Since labels are binary, when labels of those nodes are flipped, their classification accuracy may improve. When nodes with low classification accuracy are jammed, i.e., cut from the network, their potentially misleading model weights can no longer affect the network during DFL. This results in higher classification accuracy.

4.4 Joint Attacks based on Network Properties

Lastly, we build upon the correlation of node accuracies and network properties that we have established in the correlation analysis of Section 3, and consider attacks on the DFL with additional sets of attacked nodes, i.e., $TopK_d$, $TopK_C$, and $TopK_c$, where we order nodes with respect to node degree, average connected component size, and connection time ratio. We test the performance with $K = \{25, 47, 70\}$ for individual jamming and poisoning, as well as combined attack, where we choose $p_a = \{0.5, 1\}$. We provide the results in Table 7. In all three subtables, as the attack intensity increases, accuracy drops within each attack type, as expected. We also observe that poisoning attack is more impactful compared to jamming, where the joint attack lowers the accuracies even further. As also observed in Table 6, the effects of high degrees of joint jamming and poisoning attacks may cancel each other (with $p_a = 1$ across all K values), resulting in slightly higher accuracies compared to poisoning only.

Table 6: Joint effects of poisoning and jamming attacks.

(a) Average accuracy.

Jammed Nodes	$Top25_P$ Poison Prob.	
	$p_a = 0.5$	$p_a = 1$
None	0.7532	0.7378
$Top25_J$	0.7342	0.7234
$Top47_J$	0.7168	0.7040
$Top70_J$	0.6951	0.6652

(b) Minimum accuracy.

Jammed Nodes	$Top25_P$ Poison Prob.	
	$p_a = 0.5$	$p_a = 1$
None	0.6779	0.6480
$Top25_J$	0.4729	0.4740
$Top47_J$	0.5559	0.3740
$Top70_J$	0.4760	0.3310

(c) Average accuracy.

Jammed Nodes	$Top47_P$ Poison Prob.	
	$p_a = 0.5$	$p_a = 1$
None	0.7313	0.6619
$Top25_J$	0.6002	0.4837
$Top47_J$	0.5928	0.4719
$Top70_J$	0.6048	0.4726

(d) Minimum accuracy.

Jammed Nodes	$Top47_P$ Poison Prob.	
	$p_a = 0.5$	$p_a = 1$
None	0.6269	0.4189
$Top25_J$	0.3460	0.2700
$Top47_J$	0.3550	0.2390
$Top70_J$	0.3920	0.2640

(e) Average accuracy.

Jammed Nodes	$Top70_P$ Poison Prob.	
	$p_a = 0.5$	$p_a = 1$
None	0.7166	0.5314
$Top25_J$	0.7081	0.5276
$Top47_J$	0.6837	0.5622
$Top70_J$	0.6657	0.5264

(f) Minimum accuracy.

Jammed Nodes	$Top70_P$ Poison Prob.	
	$p_a = 0.5$	$p_a = 1$
None	0.5939	0.3149
$Top25_J$	0.5469	0.3490
$Top47_J$	0.4679	0.3249
$Top70_J$	0.4729	0.3170

Table 7: Effects of attacks based on network properties.

(a) Node degree.

	Jam Only		Poison Only		Joint Attack	
Attack param.	Avg.	Min	Avg.	Min	Avg.	Min
$Top25_d, p_a = 0.5$	0.7651	0.6750	0.7722	0.6769	0.7598	0.7080
$Top25_d, p_a = 1.0$			0.7204	0.4779	0.7260	0.5310
$Top47_d, p_a = 0.5$	0.7353	0.4740	0.7307	0.6430	0.7261	0.6380
$Top47_d, p_a = 1.0$			0.6407	0.3449	0.6494	0.3790
$Top70_d, p_a = 0.5$	0.6903	0.4480	0.7070	0.5950	0.6903	0.4480
$Top70_d, p_a = 1.0$			0.5360	0.3190	0.5599	0.2720

(b) Average connected component size.

	Jam Only		Poison Only		Joint Attack	
Attack param.	Avg.	Min	Avg.	Min	Avg.	Min
$Top25_C, p_a = 0.5$	0.7613	0.6570	0.7560	0.6710	0.7527	0.6589
$Top25_C, p_a = 1.0$			0.7008	0.3899	0.7188	0.4359
$Top47_C, p_a = 0.5$	0.7435	0.6520	0.7246	0.4729	0.7114	0.4760
$Top47_C, p_a = 1.0$			0.6102	0.3230	0.6339	0.3700
$Top70_C, p_a = 0.5$	0.7088	0.4740	0.7148	0.6050	0.6914	0.4650
$Top70_C, p_a = 1.0$			0.5466	0.2969	0.5434	0.3249

(c) Connected time ratio.

	Jam Only		Poison Only		Joint Attack	
Attack param.	Avg.	Min	Avg.	Min	Avg.	Min
$Top25_c, p_a = 0.5$	0.7583	0.4749	0.7471	0.6819	0.7637	0.6880
$Top25_c, p_a = 1.0$			0.7316	0.6729	0.7505	0.6869
$Top47_c, p_a = 0.5$	0.7390	0.5979	0.7340	0.6460	0.7131	0.4140
$Top47_c, p_a = 1.0$			0.6429	0.4410	0.6658	0.3799
$Top70_c, p_a = 0.5$	0.7168	0.4760	0.7066	0.4729	0.6971	0.4530
$Top70_c, p_a = 1.0$			0.5395	0.3109	0.5555	0.2879

5 CONCLUSION

We explored the distributed operation of FL for anomaly detection in inter-vehicle networks, highlighting its advantages, challenges, and vulnerabilities in the presence of multi-domain attacks such as training data poisoning and jamming. Unlike traditional FL, which relies on a central server for aggregation, DFL enables vehicles to train and exchange models locally with their one-hop neighbors, propagating knowledge through multi-hop communications. Using the VeReMi Extension Dataset, we showed that DFL improves anomaly detection compared to local-only learning, while network connectivity and data heterogeneity strongly influence model accuracy. A major concern in DFL is its vulnerability to multi-domain attacks. We showed how training data poisoning attacks propagate over multiple hops, affecting not only the initially compromised vehicles but also indirectly other ones through repeated model aggregation, and causing network-wide degradation. On the other hand, jamming attacks can disrupt model exchanges, leading to incomplete learning. We evaluated individual and joint effects of jamming and poisoning attacks. Our results highlight that ensuring adversarial robustness and resilient communication is essential for safety and reliability of AI-driven inter-vehicle networks. Future work can investigate other attacks, such as targeted poisoning and backdoor attacks, and design defense mechanisms.

ACKNOWLEDGMENTS

Research was sponsored by the Army Research Laboratory under RTX BBN Technologies, Inc. subcontract and was accomplished under Cooperative Agreement Number W911NF-24-2-0131. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

We would like to thank Stephen Raio from U.S. Army DEVCOM Army Research Laboratory for valuable feedback and guidance.

REFERENCES

[1] Ahmed Saleh Bataineh, Mohammad Zulkernine, Adel Abusitta, and Talal Halabi. 2024. Detecting Poisoning Attacks in Collaborative IDSs of Vehicular Networks Using XAI and Shapley Value. *Journal on Autonomous Transportation Systems* 2, 3 (2024).

[2] Secil Ercan, Leo Mendiboure, Lylia Alouache, Sassi Maaloul, Tidiane Sylla, and Hasnaa Aniss. 2023. An Enhanced Model for Machine Learning-Based DoS Detection in Vehicular Networks. In *IFIP Networking Conference (IFIP Networking)*.

[3] Jiaqi Huang, Yili Jiang, Sohan Gyawali, Zhiguo Zhou, and Fangtian Zhong. 2024. Semi-supervised Federated Learning for Misbehavior Detection of BSMs in Vehicular Networks. In *IEEE 100th Vehicular Technology Conference (VTC2024-Fall)*.

[4] Joseph Kamel, Michael Wolf, Rens W. van der Hei, Arnaud Kaiser, Pascal Urien, and Frank Kargl. 2020. VeReMi Extension: A Dataset for Comparable Evaluation of Misbehavior Detection in VANETs. In *IEEE International Conference on Communications (ICC)*.

[5] Qiang Lu, Hojin Jung, and Kyoung-Dae Kim. 2022. Optimization-Based Approach for Resilient Connected and Autonomous Intersection Crossing Traffic Control Under V2X Communication. *IEEE Transactions on Intelligent Vehicles* 7, 2 (2022).

[6] Yalin E Sagduyu, Tugba Erpek, and Yi Shi. 2023. Securing NextG Systems against Poisoning Attacks on Federated Learning: A Game-Theoretic Solution. In *IEEE Military Communications Conference (MILCOM)*.

[7] Y. E. Sagduyu, T. Erpek, and Y. Shi. 2025. Poisoning Attack and Defense Game for Federated Learning in Resilient NextG Networks. In *Autonomous Cyber Resilience*. Wiley-IEEE Press. to appear.

[8] Yi Shi and Yalin E Sagduyu. 2022. How to launch jamming attacks on federated learning in NextG wireless networks. In *IEEE Globecom Workshops (GC Wkshps)*.

[9] Yi Shi and Yalin E Sagduyu. 2022. Jamming attacks on federated learning in wireless networks. *arXiv preprint arXiv:2201.05172* (2022).

[10] Yi Shi, Yalin E Sagduyu, and Tugba Erpek. 2022. Federated learning for distributed spectrum sensing in NextG communication networks. *arXiv preprint arXiv:2204.03027* (2022).

[11] Yi Shi, Yalin E Sagduyu, and Tugba Erpek. 2023. Jamming attacks on decentralized federated learning in general multi-hop wireless networks. In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*.

[12] Chen-Khong Tham, Lu Yang, Akshit Khanna, and Bhavya Gera. 2023. Federated learning for anomaly detection in vehicular networks. In *IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*.

[13] Zhe Wang and Tingkai Yan. 2023. Federated learning-based vehicle trajectory prediction against cyberattacks. In *IEEE 29th International Symposium on Local and Metropolitan Area Networks (LANMAN)*.

[14] Hadi Yakan, Ilhem Fajjari, Nadjib Aitsaadi, and Cedric Adjih. 2023. Federated learning for V2X misbehavior detection system in 5G edge networks. In *ACM Conference on Modeling Analysis and Simulation of Wireless and Mobile Systems.*