Journal of Multimedia Information System VOL. 11, NO. 1, March 2024 (pp. 57-66): ISSN 2383-7632 (Online)
https://doi.org/10.33851/JMIS.2024.11.1.57

Federated Learning-Based Detection and Control Mechanism of In-Car Navigation Safety System

Jingge Gao[1], Shuqiang Zhang[2*], Wei Lu[1]

Abstract

In-car navigation systems face challenges in ensuring safety while preserving data privacy. This paper proposes PrivNav, a federated learning scheme integrating differential privacy and secure multi-party computation for privacy-preserving learning. PrivNav enables vehicles to collaboratively train a model by aggregating locally computed updates without sharing raw data. Perturbations and secret sharing protect sensitive information and prevent inference attacks. PrivNav outperforms existing federated learning schemes by accommodating user dropouts, supporting customizable aggregation methods beyond FedAvg, and extending to decentralized scenarios without a trusted authority. Experiments demonstrate PrivNav's strong privacy guarantees and high accuracy, significantly enhancing the detection and control capabilities of in-car navigation safety systems. Precise event detection, abnormal situation differentiation, and reduced false alarms are achieved, improving overall system safety, trust, and performance.

Key Words: In-Car Navigation, Federated Learning, Deep Learning, Differential Privacy.

I. INTRODUCTION

Recently, in-car navigation systems have become integral to modern vehicles, providing drivers with real-time guidance and enhancing their overall driving experience. These systems rely on GPS technology, map data, and intelligent algorithms to offer accurate and efficient navigation [1-2]. However, the increasing complexity and reliance on in-car navigation systems pose significant challenges in ensuring their safety and reliability. Traditional approaches to detecting and controlling safety issues in these systems often involve centralized analysis, which may compromise data privacy and result in performance limitations [3]. Therefore, there is a need for a decentralized mechanism that can enhance the accuracy and efficiency of safety detection and control in in-car navigation systems while preserving data privacy [4-5]. Ensuring the safety of in-car navigation systems is paramount to preventing accidents and mitigating potential hazards. Traditional approaches to safety mechanisms have relied on centralized systems that analyze data from individual vehicles. However, this approach raises concerns about data privacy and scalability. Additionally, centralization may result in performance limitations and delays in detecting and responding to safety issues.

Federated learning is a privacy-preserving approach that enables the collaborative training of machine learning models across a network of devices or vehicles without sharing raw data [6-7]. In the context of in-car navigation safety systems, each vehicle possesses a dataset consisting of sensor readings, location information, and other relevant data collected during driving. Instead of transmitting the raw data to a central server for analysis, federated learning allows the vehicles to perform local model training using their respective datasets. The training process involves iteratively updating the model parameters based on the local data. The updated model parameters are then sent to a central server, aggregating them with the parameters from other vehicles [8]. This aggregation step is crucial as it combines the knowledge learned from different vehicles without exposing their specific data [9]. Various techniques, such as secure aggregation protocols and encryption, can be employed to preserve privacy during the model aggregation process. By leveraging the collective intelligence of a network of vehicles, the resulting model captures the common patterns and insights from diverse driving scenarios, enhancing the overall detection and control mechanisms of the in-car navigation safety system. The decentralized nature

Manuscript received October 16, 2023; Revised December 21, 2023; Accepted December 27, 2023. (ID JMIS-23M-10-044)
Corresponding Author (*): Shuqiang Zhang, +86-17731020378, 17731020378@163.com
[1]School of Information and Electrical Engineering, Hebei University of Engineering, Handan, China, Gjg19760405@126.com, 13722370914@163.com
[2]Department of Software and Big Data, Handan Polytechnic College, Handan, China, 17731020378@163.com

of federated learning ensures that sensitive information, such as specific driving routes or personally identifiable information, remains on local devices and is not shared with external parties [10].

This study suggests an effective federated learning approach that protects privacy. Based on the multi-server-multi-client architecture, the client downloads the global model, trains with local data, adds noise to the updated parameters obtained by training, and secretly shares them with all servers. The servers perform secure multi-party computation based on the shared shares and obtain the shared shares of the aggregate result. The client downloads all the shares, recovers the aggregate results, and updates the model. After analysis, this method protects privacy, tolerates dropped calls, is compatible with various aggregation functions, and is easy to extend to decentralized scenarios. The proposed effective federated learning approach that protects privacy provides significantly enhanced privacy preservation through formal differential privacy guarantees compared to existing federated learning techniques. It demonstrates greater robustness and flexibility via custom aggregation methods, user dropout resilience, and decentralization capabilities.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 presents the methodology. The simulation and results analysis are presented in Section 4, and Section 5 concludes the paper.

II. RELATED WORKS

2.1. In-Car Navigation Systems

In-car navigation systems are advanced technological solutions integrated into vehicles to provide drivers with real-time guidance and assistance. These systems utilize a combination of hardware, software, and data to offer drivers accurate directions, map displays, and various features to enhance their driving experience. Global positioning system (GPS) technology forms the foundation of in-car navigation systems [11]. GPS receivers in vehicles receive signals from satellites to determine the vehicle's precise location on Earth. This information is then used to provide accurate navigation guidance. In-car navigation systems rely on detailed map data, including road networks, landmarks, points of interest (POIs), and traffic information [12-13]. In-car navigation systems often provide voice-guided instructions to drivers, ensuring hands-free operation and minimizing distractions. Voice prompts guide drivers through turns, lane changes, and other maneuvers, enhancing safety and convenience [14-15]. In-car navigation systems calculate optimal routes based on the selected destination, considering traffic conditions, road closures, and real-time data. Traffic information can be sourced from various providers, including GPS probes, sensors, and crowd-sourced data. Users can search for specific POIs or browse categories to find relevant services [16].

2.2. Federated Learning in Safety Systems

Federated learning is a machine learning approach that has gained significant attention in developing safety systems, including in-car navigation safety systems. It offers unique advantages for preserving data privacy and improving the accuracy and reliability of safety mechanisms [17-20]. Federated learning enables multiple vehicles or devices to collaborate in training a shared machine learning model without sharing their raw data. Each vehicle locally trains the model using its dataset, which consists of relevant sensor readings, location information, and safety-related data. The training process occurs on the individual vehicles, ensuring that sensitive information remains on the device and is not exposed to external parties. This decentralized approach enhances data privacy and addresses concerns about sharing personal driving data [21-22]. Once the local training is complete, the updated model parameters are securely aggregated without revealing the specific data from each vehicle. Aggregation methods such as secure multi-party computation or differential privacy techniques can be used to ensure privacy during the model parameter merging process [23]. Combining the knowledge from multiple vehicles, the resulting model captures a comprehensive understanding of navigation safety patterns, including potential hazards, driving behaviors, and road conditions [24]. Federated learning facilitates the development of robust hazard detection mechanisms in in-car navigation safety systems.

III. PROPOSED MECHANISM

3.1. General Idea and Framework

Federated learning allows data nodes to perform multiple rounds of local model training locally and then upload the local model to a central node for parameter aggregation, thus avoiding transmitting the original data across nodes and protecting data privacy to a certain extent. The core idea of the FedAvg algorithm is to use intermediate information, such as model parameters, to replace the original data to transmit between nodes [25]. However, this intermediate information is often the "refinement" of the knowledge contained in the original data, and there is still a risk of privacy leakage when exposed to adversaries. In this paper, privacy leakage is mainly divided into two categories. (i) Privacy leakage caused by local information exposure. (ii) Privacy leakage caused by global information exposure.

This paper proposes the following scheme ideas to resist two types of privacy leakage risks. (i) The adversary can reconstruct the local dataset from the data uploaded by a

client in each round. In this paper, we use secure multi-party computation to hide the data uploaded by the client in each round and ensure that the server can summarize the uploaded data to obtain the correct aggregation results to avoid the leakage of local information. (ii) Considering that the amount of information in the data only decreases with the computation or processing, when the adversary cannot steal the personal data uploaded by the user, the closest information to the original data can be obtained is the aggregation model in each round. In this paper, based on the idea of local differential privacy, the client adds a specific perturbation to the local model obtained by local training and uploads the perturbed model to the server so that the aggregation process of each round satisfies differential privacy, that is, whether a sample of a client participates in the training or not, the distribution of the global model after aggregation does not change significantly [26]. Thus, the aggregation model can be prevented from being exploited by adversaries. For simplicity, we name the proposed federated learning scheme PrivNav, which stands for privacy-preserving navigation federated learning.

The overall framework of PrivNav is shown in Fig. 1. Participating nodes include (i) n clients $C_1, C_2, \cdots, C_n$, responsible for local storage of their private datasets. (ii) m servers $S_1, S_2, \cdots, S_m$, $m \geq 2$, responsible for aggregate calculation of data shares. There is a secure channel between the client and the server. Table 1 lists some notations and descriptions used in this paper.

3.2. Threat Model

The system mainly has three types of roles: client, server, and external adversary. This paper mainly considers the first two types of internal adversaries who directly participate in the training process and are more threatening.

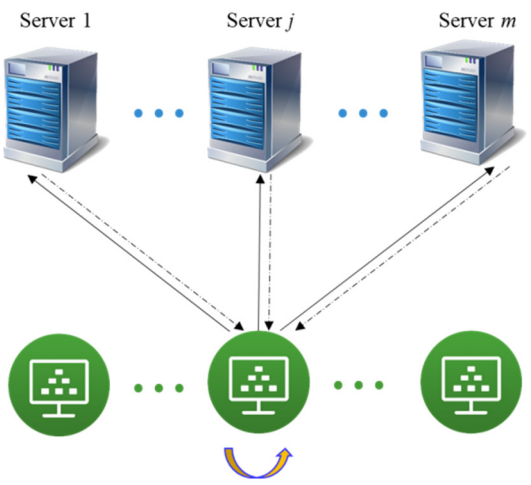

Fig. 1. Framework overview of PrivNav.

Table 1. Symbols and descriptions.

Symbol	Description
S_i	ith server node
C_i	ith client node
D_i	Local dataset of C_i
$\|D_i\|$	The number of samples that D_i contains
N	Minimum of $\|D_i\|$
M^r	Global model of rth round
M_i^r	Local model of the client C_i in the rth round
$M_{i,j}^r$	Model shares uploaded by C_i to S_j in the rth round
$M_{*,j}^r$	Aggregate share of server S_j in the rth round
R	Total number of training rounds
C	Upper bound on the L2 norm
K	Lower bound on the number of clients
B	Size of mini-batch
E	The number of iterations of the client traverses the dataset

Server. Assume that the server is semi-honest. It can correctly execute the algorithm and protocol process, but it will try to infer more private information based on the collected data. Simultaneously, it is assumed that the number of colluding server opponents is less than the threshold t of secret sharing, with (n, n)-threshold secret sharing scheme as an example, assuming that there is at least one honest server.

Client. Assuming that the client is semi-honest, the goal of the adversary client is to obtain the relevant information of the honest client's training data by viewing the interactive content rather than uploading maliciously tampered data that will reduce the accuracy of the model or even cause the training not to converge. Simultaneously, the number of colluding clients is assumed to be less than $n-1$. Otherwise, for the reversible aggregation function $F(d_1, \cdots, d_n)$, the colluding node can infer the input of the only honest node through the output and the known $n-1$ inputs.

External adversary. The model is deployed to a node or cloud to provide prediction services after training. The adversary can analyze the output from limited access to the model interface and try to infer local data on a client. Considering the knowledge and ability of the adversary, the external adversary cannot obtain the intermediate information of the training process, so the attack's success rate is often lower than the above two types of internal adversaries.

3.3. Training Process

The algorithm incorporates the following parameters: a set of servers denoted as $S = \{S_1, S_2, \cdots, S_m\}$, where m is

greater than or equal to 2, and a set of clients represented by $C = \{C_1, C_2, \cdots, C_n\}$, with n being greater than or equal to 3. Each client corresponds to a local dataset, $D = \{D_1, D_2, \cdots, D_n\}$. It is assumed that a minimum number of K clients upload parameters per round. The machine learning algorithm employed, denoted as L, is consistently executed by all clients during their local training. In this study, the optimization algorithm utilized to train the model M is gradient descent, with the model architecture declared prior to the training process. The primary focus of this research lies in training neural networks. The parameters for differential privacy are ε and δ, where smaller values correspond to higher degrees of privacy protection. The maximum number of colluding servers is denoted as t', and the threshold of the secret sharing scheme should exceed this value. Lastly, the total number of training rounds is denoted as R.

The specific procedure of the algorithm for training a privacy-preserving federated learning model referred to as Algorithm 1, is presented as follows within an academic context. Initially, the server initializes the model parameters, denoted as S_1. Subsequently, the client downloads the model and employs its local dataset for training, which results in acquiring new model parameters. A sequence of operations is conducted on the local model to maintain control over the sensitivity of the aggregated model parameters. Initially, the local model is trimmed and compressed, followed by the addition of qualified noise. The resulting model is then shared among all servers in a secretive manner. In this context, the FedAvg weighted average technique aggregates the model parameters. To facilitate clarity and simplicity in the algorithm presentation, each client is assumed to employ a dataset of equal size for local training. After a specific duration, allowing for adequate parameter updates, the server locally averages the parameter shares to obtain the aggregated shares. The client subsequently downloads the aggregate shares from each server to reconstruct the secret and obtain the updated model parameters. Repeating these steps can inform the training process until the desired objective is achieved.

Secret sharing and secure computation protocols are commonly designed based on algebraic structures like finite fields or commutative rings. However, these structures are not directly applicable to real-world data scenarios. Consequently, it becomes essential to appropriately encode data and establish a mapping relationship with the aforementioned algebraic structures. In PrivNav, we transfer the model parameters to the ring Z_2^l, where fixed-point numbers with l bits represent the actual parameters. Within this representation, the lower e bits are allocated for the decimal places. To illustrate, consider the floating-point parameter x in Step14 of the local model M_i^r. Its encoded form, denoted as x', is obtained using the expression $x' = \text{int}(x \times 2^e)$, where int denotes the rounding operation. In this study, we set l to be 64 and e to be 32, allowing us to store the encoded data using the int64 data type. Notably, the encoded fixed-point number exhibits a maximum range of expression defined as $[-2^{l-e-1} + 2^{-e}, 2^{l-e-1} - 2^{-e}]$. On the other hand, given an encoded value x', the decoding process involves a simple computation of $x = x'/2^e$, enabling the retrieval of the original parameter.

Consider two numbers, x, and y, that are shared among n nodes, denoted as $[x] = \{x_1, x_2, \cdots, x_n\}$ and $[y] = \{y_1, y_2, \cdots, y_n\}$, where each node i possesses x_i and y_i. To compute the share of $x + y$, each node independently computes $x_i + y_i$. Consequently, in Step19 of Algorithm 1, S_i adds the local shares to obtain the sum of the local models. It is also straightforward to observe that constant multiplication is performed locally. Given a share $[x]$ and a constant c, each node can locally compute $c \times x_i$ to obtain the share $[cx]$. The $\sum_{i=1}^{n} c \times x_i$ can be obtained

Algorithm 1. Privacy-preserving federated learning model trai ning (PrivNav).

Input: Machine learning algorithm L, client set $C = \{C_1, C_2, \cdots, C_n\}$, local dataset $D = \{D_1, D_2, \cdots, D_n\}$, server set $S = \{S_1, S_2, \cdots, S_m\}$, number of server nodes $t'(2 \le t' < m)$ that can tolerate collusion, minimum number of clients K for uploading para meters per round, total number of training rounds R

Output: Trained model M

01: S_1 initializes the global model M
02: **for** $r \leftarrow 1$ to R
03: **for** $C_i \in C$ do
04: **if** $r = 1$
05: Download the initial global model M from S_1
06: **else**
07: Download the share $M_{*,j}^{r-1}$ from $S_{i,j} \in \{1,2,\cdots, m\}$
08: $M^{r-1} \leftarrow \text{SecRec}\left(M_{*,1}^{r-1}, \cdots, M_{*,m}^{r-1}\right)$
09: **end-if**
10: local training $M_i^r \leftarrow (M^{r-1}, D_i)$
11: clipping weights $M_i^r / \max(1, |M_i^r|/C)$
12: adding noise $M_i^r \leftarrow M_i^r + \text{noise}(\varepsilon, \delta, C, K)$
13: $t \leftarrow t' + 1$
14: computing shares $\left(M_{i,1}^r, \cdots, M_{i,m}^r\right) \leftarrow \text{SecShr}(M_i^r, m, t)$
15: send $M_{i,j}^r$ to $S_{i,j}$
16: **end-for**
17: **for** $S_j \in S$ **do**
18: wait until enough parameters are collected to update the parameters
19: aggregate share $M_{*,j}^r \leftarrow \left(M_{i_1,j}^r + \cdots + M_{i_K,j}^r\right)/K$
20: **end-for**
21: **end-for**
22: download shares $M_{*,j}^R$, recover M^R
23: output M^R

through secret recovery as $c\sum_{i=1}^{n} x_i$, which ultimately yields cx. Notably, since the parameter K in Step19 is a constant value for the server, the computation for the average share can also be carried out locally.

The resilience of Algorithm 1 to client disconnection is evident. Considering that the participating data nodes in the training process often consist of unstable mobile edge devices, the privacy protection scheme must ensure the effectiveness of the training process even when nodes experience periods of disconnection. In the context of Algorithm 1, if a client becomes disconnected, it results in the absence of the share of the model update being sent to the server. In such scenarios, the client is treated as non-participating in the current round of training. Notably, since all shares are simultaneously transmitted to all servers after executing the SecShr algorithm, the algorithm assumes that no client can selectively send shares to specific servers. However, if such a situation occurs, the server can efficiently resolve it by conducting an additional round of communication. This additional round would confirm the source client IDs for all received shares and subsequently intersect them when performing the aggregation process.

Additionally, PrivNav demonstrates compatibility with more intricate custom aggregation functions. While FedAvg utilizes a weighted average as the aggregation operation for parameters, more is needed to meet the demands of complex application requirements. For instance, to combat Byzantine attacks, some researchers have proposed the computation of the median among all client update values, which is then employed as the aggregation result. Unlike privacy-preserving schemes based on homomorphic encryption or function encryption that solely support linear aggregation operations PrivNav enables the computation of any complex aggregation function $g(x_1, x_2, \cdots, x_n)$. This is achieved by redefining Step19 of PrivNav as $M_j^r \leftarrow \mathrm{SecComp}(g(M_{i_1,j}^r, \cdots, M_{i_K,j}^r))$, where the secure multi-party computation protocol SecComp may introduce additional communication among servers. The volume and number of communication rounds in SecComp are influenced by the specific aggregation function g.

Lastly, a trusted center is often relied upon in existing federated learning frameworks. However, in this paper, using secure multi-party computation for parameter aggregation naturally extends to decentralized scenarios. In such scenarios, each party serves as a data node and a computation node, conducting local model training and assuming responsibility for secure parameter aggregation. Specifically, the parties involved are denoted as $\{C_1, C_2, \cdots, C_n\} = \{S_1, S_2, \cdots, S_m\}$, where m equals n. Adopting a (n, n)-threshold secret sharing scheme ensures that each party is not required to place trust in other participants, and their respective parameters cannot be reconstructed.

IV. SIMULATION AND RESULTS ANALYSIS

This study primarily focuses on analyzing and evaluating PrivNav based on three key aspects: privacy, efficiency, and usability. To conduct the experiments, each client or server within the scheme is assigned an experimental node consisting of an 8-core / 32GB cloud host instance. All nodes are configured to operate within the same subnet. As the accuracy of the resulting model is independent of whether the nodes are situated in an actual distributed environment, a simulated federated learning training process involving multiple nodes is executed on a 16-core / 64GB cloud host. Local model training is implemented using PyTorch, while secure multi-party computation, differential privacy, and inter-node communication are implemented using Python3. The MNIST dataset is selected as both the training and test sets and a convolutional neural network architecture is adopted for the training model. Specifically, the model comprises two convolutional layers with a 5×5 convolution kernel size, 10 and 20 output channels, a stride of 1, and valid padding. A 2×2 Max pooling layer and ReLU activation function are applied following the convolutional layers. Lastly, the model consists of two fully connected layers with dimensions of (320, 50) and (50, 10), respectively. Additionally, a dropout of 0.5 is applied after the second convolutional layer and the first fully connected layer [27]. Fig. 2 visually represents the model architecture and a single forward computation process.

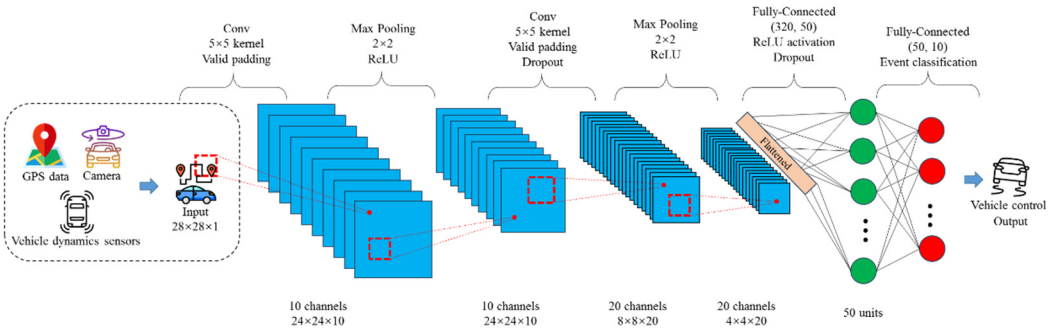

Fig. 2. Convolutional neural network architecture for in-car navigation system.

4.1. Efficiency Analysis

The practicality of a federated learning scheme hinges on its operational efficiency. This study employs secure multi-party computation utilizing secret sharing to safeguard the security of the aggregation process. Furthermore, differential privacy is employed to ensure the security of the aggregation results. However, it should be noted that the transmission of shared shares introduces additional communication overhead. Additionally, the steps involved in secret sharing, share calculation, secret recovery, model pruning, and noise addition introduce additional computational overhead.

To assess the practical execution efficiency of PrivNav, the FedAvg algorithm, eecFed [21], and MLFL [22] were employed as a benchmarks for comparison. The objective was to demonstrate that PrivNav does not suffer from significant efficiency losses while enhancing privacy. Fig. 3 illustrates the variation in cumulative time consumption of PrivNav, FedAvg, eecFed, and MLFL as the number of rounds increases. The experiments were conducted with different local dataset sizes for each client, namely 600, 3,000, and 6,000. The remaining parameters were set as follows: $B = \infty$, $E = 3$, $n = k = 10$, $m = 2$, and $R = 100$.

More efficient rounds of communication allow aggregated models to be updated and deployed to vehicles more rapidly, enabling improved real-time detection and response to emerging hazards. By reducing communication and computation loads, PrivNav allows resource-constrained in-car systems to dedicate more processing and bandwidth to safety-critical applications. The linear scalability demonstrated by PrivNav supports exponential growth in connected vehicles without efficiency degradation, ensuring seamless safety systems as adoption increases. Additionally, more efficient model retraining cycles facilitate faster improvement and adaptation of automated vehicle control policies to address new hazards detected by updated models. PrivNav's efficiency optimizations translate to quicker security updates, lower infrastructure burden, seamless scalability, and rapid control iteration as in-car navigation safety systems expand, directly strengthening detection precision, response times, and overall system robustness.

With the growing size of the client dataset, the disparity in efficiency between PrivNav and FedAvg remains the same. This reduction can be attributed to the fact that as the client dataset expands, the increase in local training time becomes more significant compared to the communication time. Similarly, augmenting each client's local training rounds (E) further narrows the efficiency gap between the two methods. Efficiency tests were conducted on more complex datasets and models to validate these assertions. The Cifar100 dataset was employed as the training dataset, while ResNet-50 served as the training model [28]. The parameters were set as follows: $B = \infty$, $E = 3$, $n = k = 10$, $m = 2$, and $R = 100$. The parameter values for N and E were deliberately reduced to accentuate the efficiency disparity between PrivNav and FedAvg. The results are illustrated in Fig. 4. Notably, when there are merely 250 training samples in each client, the discrepancy in efficiency between the two methods is distinctly observable. However, as N increases to 500 and 1,000, the efficiency of the two methods becomes highly comparable.

Subsequently, the scalability of PrivNav is evaluated by examining the efficiency variations under different scenarios involving varying numbers of clients and servers. In these tests, each client's local dataset is held constant at 3,000 samples, while the other parameters remain consistent with the settings depicted in Fig. 3. The experimental outcomes are presented in Fig. 5 to demonstrate the efficiency changes.

Fig. 5(a) illustrates the average time consumption per round as the number of clients increases, with the number

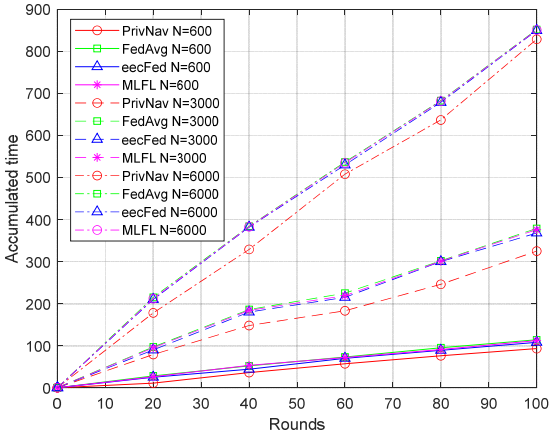

Fig. 3. Accumulated training time of two approaches under three different datasets.

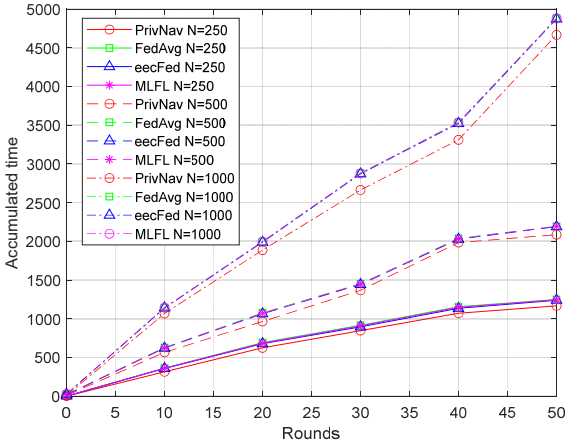

Fig. 4. Efficiency comparison of two approaches under Cifar100 + ResNet-50 setting.

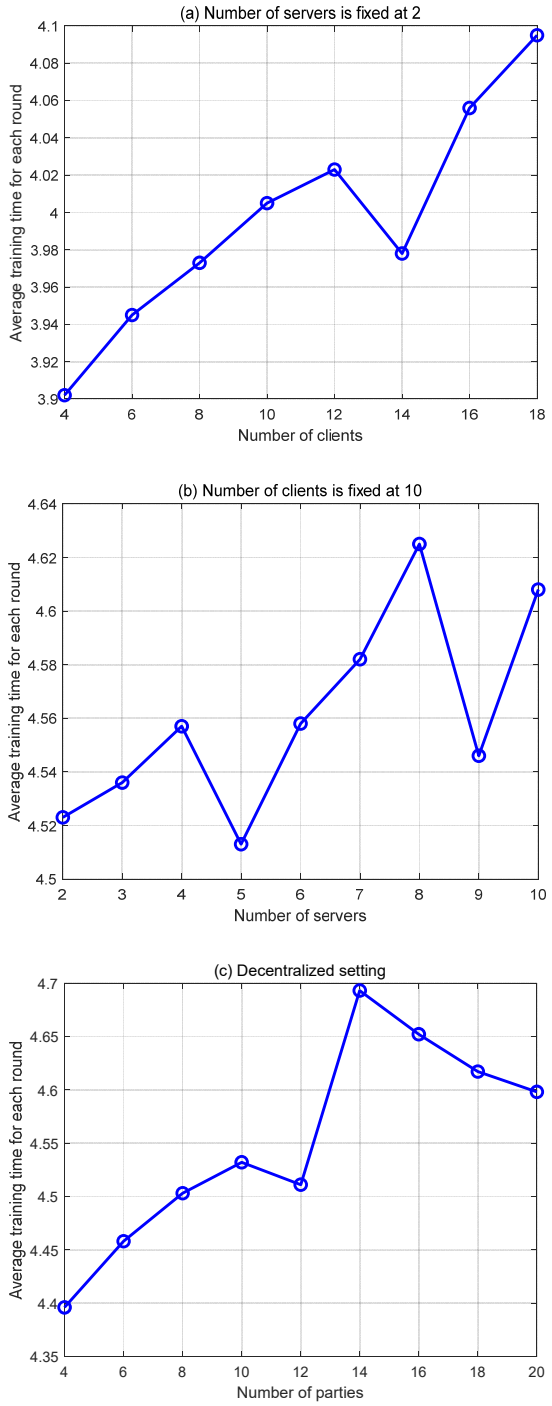

Fig. 5. Average training time for each round as number of clients/servers increases.

of servers fixed at 2. The graph shows that, apart from fluctuations caused by network conditions, the average time consumption per round of PrivNav grows linearly with the number of clients. The rate of increase is approximately 0.0125 (seconds per client), indicating that the overall efficiency of the scheme remains within an acceptable range as the number of participating nodes increases. Fig. 5(b) demonstrates the average time consumption per round when the number of clients is fixed at ten, and the number of servers varies. In this experiment, the number of clients is not less than the number of servers ($n \geq m$), and the network conditions of both clients and servers are similar. Fig. 5(c) illustrates that the average time consumption does not change significantly compared to the centralized scenario. As the number of participating nodes increases, the average time consumption remains relatively stable.

The efficiency of PrivNav plays a crucial role in the detection and control of in-car navigation safety systems. These systems necessitate timely and accurate detection and control mechanisms to respond to potential hazards effectively. The efficiency of the federated learning scheme directly impacts the speed at which the aggregated model can be updated and deployed to the in-car navigation systems. A more efficient scheme enables faster model updates, facilitating quicker detection and control of safety-related events. Efficient federated learning is particularly significant for in-car navigation systems due to their limited computing resources and operation under constrained network conditions. By optimizing the efficiency of the federated learning process, the scheme reduces the computational and communication overhead associated with model aggregation and updating. This optimization allows the system to utilize available resources effectively without excessive strain. PrivNav strongly emphasizes privacy preservation to safeguard sensitive data collected from in-car navigation systems during the learning process. By executing the federated learning algorithm efficiently, the scheme minimizes the exposure of raw data to external entities, thereby mitigating privacy risks. Consequently, users' trust and confidence in the system are enhanced, fostering active participation. In-car navigation safety systems operate in large-scale environments with numerous interconnected vehicles. The efficiency of the federated learning scheme influences its scalability to accommodate an increasing number of clients and servers. A highly efficient scheme can handle a more extensive system's growing computational and communication demands, ensuring seamless and effective detection and control across various vehicles. An efficient federated learning scheme facilitates real-time responsiveness, optimal resource utilization, enhanced privacy preservation, and scalability. These factors collectively contribute to the effectiveness and reliability of the detection and control mechanisms in in-car navigation safety systems.

4.2. Usability Analysis

The model's accuracy plays a crucial role in determining the usability of a federated learning scheme. In secure multi-party computation, computations are performed over finite fields or commutative rings. However, user data is typically represented using fixed-point numbers, requiring

truncation during the computation process. Additionally, introducing noise due to differential privacy mechanisms can also impact the accuracy performance of the model. This subsection focuses on conducting experiments to evaluate PrivNav's influence on the model's accuracy. Specifically, for each client, a local dataset consisting of 6,000 randomly selected samples is used ($N = 6{,}000$). The remaining parameters are set as follows: $B = \infty$, $E = 3$, $C = 10$, $n = k = 100$, $m = 2$, $R = 100$, and $\delta = 0.0001$. These experimental settings allow for assessing how PrivNav affects the model's accuracy in a controlled environment.

Table 2 presents the model test accuracy of both FedAvg and PrivNav after 100 rounds of communication, considering different privacy settings with overall privacy budgets of 1 and 0.5. Notably, when no noise is added, and data truncation is performed, the model's accuracy remains unaffected. As the privacy parameter ε decreases, the degree of privacy protection the learning algorithm provides improves, resulting in more significant amounts of added noise. Consequently, the model's accuracy gradually decreases, and even the convergence of the model may be impacted. Fig. 6 provides insights into this relationship, demonstrating that when ε is less than 0.0005 and the noise level (δ) exceeds 0.29, the model's prediction accuracy is notably poor. Excessive noise significantly hinders the typical iteration of the model, leading to a failure in achieving convergence. However, within the 0.0005< ε<0.0006, the model performance demonstrates significant improvement. As ε increases beyond 0.003, the model gradually stabilizes and attains the desired effect, indicating that the added noise no longer substantially hinders the model's convergence.

PrivNav's high accuracy significantly impacts the detection and control of in-car navigation safety systems. In-car navigation safety systems rely on accurate and reliable detection mechanisms to identify potential hazards and ensure timely control actions. By achieving high model accuracy through the federated learning scheme, these systems' detection and control capabilities are greatly enhanced. A high accuracy model in the federated learning scheme enables more precise and reliable predictions, improving the system's ability to detect safety-related events such as collisions, obstacles, or hazardous road conditions. Accurate detection allows the system to respond promptly, triggering

Table 2. Comparison of different approaches on model prediction accuracy.

Methods	Test accuracy
FedAvg	97.12
PrivNav	97.10
PrivNav ($\varepsilon = 0.01$)	96.59
PrivNav ($\varepsilon = 0.005$)	95.99

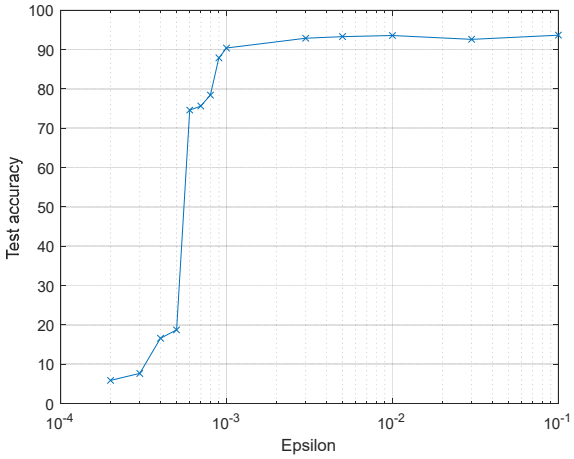

Fig. 6. Test accuracy under different privacy-preserving level.

appropriate control actions to mitigate or avoid potential risks. This contributes to enhancing overall safety for passengers and vehicles on the road. Moreover, the high accuracy of the model enhances the system's ability to differentiate between normal driving conditions and abnormal or anomalous situations. This is particularly important for identifying critical events that require immediate attention, such as sudden lane departures, aggressive driving behaviors, or potential mechanical failures. By accurately detecting such events, the system can activate appropriate control mechanisms, such as issuing warnings or adjusting vehicle settings, to ensure safe operation and prevent accidents. Additionally, the high accuracy of the federated learning scheme improves the reliability of the system's predictions, reducing false positives and false negatives. This minimizes the occurrence of unnecessary control interventions or missed detection of actual safety threats, leading to a more efficient and effective overall detection and control process. The impact of high accuracy extends beyond the detection and control mechanisms themselves. It also fosters user trust and confidence in the in-car navigation safety system. When users have confidence in the system's ability to detect and respond to safety-related events accurately, they are more likely to rely on the system and follow its recommendations. This promotes greater user acceptance and utilization of the system, leading to improved overall safety outcomes. In conclusion, the high accuracy achieved through PrivNav significantly enhances in-car navigation safety systems' detection and control capabilities. It enables precise and reliable event detection, differentiation of abnormal situations, and reduces false alarms, improving overall safety, user trust, and system performance.

V. CONCLUSION

In in-car navigation systems, this study provides a federated learning strategy for deep learning that protects privacy.

To protect the privacy of local data and computing processes, the suggested scheme integrates secure multi-party computation with differential privacy techniques. By adding perturbations to local models and securely sharing them with central servers, the scheme prevents unauthorized access to sensitive information and malicious inference from shared data. The scheme also accommodates user dropouts and supports various aggregation functions. Furthermore, it can be extended to decentralized scenarios, eliminating the need for a trusted central authority. The experimental results demonstrate the effectiveness of PrivNav in preserving privacy and enhancing the accuracy of in-car navigation systems. By emphasizing privacy preservation, sensitive data collected from in-car navigation systems is safeguarded during learning. The high accuracy achieved through the federated learning scheme significantly improves these systems' detection and control capabilities. It enables precise and reliable event detection, differentiation of abnormal situations, and reduces false alarms, ultimately enhancing overall safety, user trust, and system performance.

While there are limitations to consider, first, PrivNav assumes the availability of a reliable and secure communication infrastructure. The efficiency and performance of the scheme may be affected in scenarios with limited network resources or high latency. Second, the scheme's scalability should be further investigated, especially when dealing with many participants or complex datasets. The scheme's robustness against sophisticated attacks and adversarial scenarios also requires further exploration. In future research, addressing these limitations and exploring potential enhancements is essential. This could involve investigating communication-efficient protocols to improve the scheme's performance under constrained network conditions. Moreover, exploring techniques to enhance the scalability of the scheme and handle larger-scale deployments will be beneficial. Additionally, advancing the security aspects of the scheme to withstand adversarial attacks and ensuring robustness will be crucial for real-world applications.

REFERENCES

[1] I. Skog and P. Handel, "In-car positioning and navigation technologies: A survey," *IEEE Transactions on Intelligent Transportation System,* vol. 10, no. 1, pp. 4-21, 2009.

[2] M. L. Galvao, J. Krukar, and A. Schwering, "Evaluating schematic route maps in wayfinding tasks for in-car navigation," *Cartography and Geographic Information Science*, vol. 48, no. 5, pp. 449-469, 2021.

[3] P. Q. Lin, C. H. Zhou, and Y. Cheng, "A systematic cooperation method for in-car navigation based on future time windows," *Promet –Traffic & Transportation*, vol. 34, no. 3, pp. 381-396, 2022.

[4] M. Arulprakash and R. Jebakumar, "People-centric collective intelligence: Decentralized and enhanced privacy mobile crowd sensing based on blockchain," *Journal of Supercomputing*, vol. 77, no. 11, pp. 12582-12608, 2021.

[5] A. Y. Lin and Q. Ling, "Decentralized and privacy-preserving low-rank matrix completion," *Journal of the Operations Research Society of China*, vol. 3, pp. 189-205, 2015.

[6] K. Zhang, X. Song, C. Zhang, and S. Yu, "Challenges and future directions of secure federated learning: A survey," *Frontiers of Computer Science*, vol. 16, p. 165817, 2022.

[7] H. Ratnayake, L. Chen, and X. Ding, "A review of federated learning: Taxonomy, privacy and future directions," *Journal of Intelligent Information Systems*, vol. 61, no. 3, pp. 923-949, 2023.

[8] R. Gupta and T. Alam, "Survey on federated-learning approaches in distributed environment," *Wireless Personal Communications*, vol. 125, no. 2, pp. 1631-1652, 2022.

[9] K. Pillutla, Y. Laguel, J. Malick and Z. Harchaoui, "Federated learning with superquantile aggregation for heterogeneous data," *Machine Learning*, pp. 1-68, 2023.

[10] A. Qammar, A. Karim, H. Ning, and J. Ding, "Securing federated learning with blockchain: A systematic literature review," *Artificial Intelligence Review*, vol. 56, no. 5, pp. 3951-3985, 2023.

[11] W. Hebblewhite and A. J. Gillett, "Every step you take, we'll be watching you: Nudging and the ramifications of GPS technology," *AI & Society*, vol. 36, pp. 863-875, 2021.

[12] E. Park and K. J. Kim, "Driver acceptance of car navigation systems: Integration of locational accuracy, processing speed, and service and display quality with technology acceptance model," *Personal and Ubiquitous Computing*, vol. 18, pp. 503-513, 2014.

[13] C. K. Allison and N. A. Stanton, "Constraining design: applying the insights of cognitive work analysis to the design of novel in-car interfaces to support eco-driving," *Automotive Innovation*, vol. 3, no. 1, pp. 30-41, 2020.

[14] H. W. Gierlich, "Voice recognition and in-car communication testing procedures and performance parameters," *ATZextra Worldwide*, vol. 23, no. 2, pp. 32-37, 2018.

[15] A. Biswas, P. K. Sahu, and M. Chandra, "Multiple cameras audio visual speech recognition using active appearance model visual features in car environment," *International Journal of Speech Technology*, vol. 19,

pp. 159-171, 2016.
[16] A. Psyllidis, S. Gao, Y. Hu, E. K. Kim, G. McKenzie, and R. Purves, et al., "Points of interest (POI): A commentary on the state of the art, challenges, and prospects for the future," *Computational Urban Science*, vol. 2, no. 1, p. 20, 2022.
[17] D. Sirohi, N. Kumar, P. S. Rana, S. Tanwar, R. Iqbal, and M. Hijjii, "Federated learning for 6G-enabled secure communication systems: A comprehensive survey," *Artificial Intelligence Review*, vol. 56, no. 10, pp. 11297-11389, 2023.
[18] Q. Yang, A. Huang, L. Fan, C. S. Chan, J. H. Lim, and K. W. Ng, "Federated learning with privacy-preserving and model IP-right-protection," *Machine Intelligence Research*, vol. 20, no. 1, pp. 19-37, 2023.
[19] G. Han, T. Zhang, Y. Zhang, G. Xu, J. Sun, and J. Cao, "Verifiable and privacy preserving federated learning without fully trusted centers," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 3, pp. 1431-1441, 2022.
[20] J. Wen, Z. Zhang, Y. Lan, Z. Cui, and W. Zhang, "A survey on federated learning: challenges and applications," *International Journal of Machine Learning and Cybernetics*, vol. 14, no. 2, pp. 513-535, 2023.
[21] Y. Wang, Y. Tian, X. Yin, and X. Hei, "A trusted recommendation scheme for privacy protection based on federated learning," *CCF Transactions on Networking*, vol. 3, no. 3, pp. 218-228, 2020.
[22] A. Boualouache and T. Engel, "Federated learning-based scheme for detecting passive mobile attackers in 5G vehicular edge computing," *Annals of Telecommunications*, vol. 77, no. 3, pp. 201-220, 2022.
[23] J. Liu, J. Huang, Y. Zhou, X. Li, S. Ji, and H. Xiong, "From distributed machine learning to federated learning: A survey," *Knowledge and Information Systems*, vol. 64, no. 4, pp. 885-917, 2022.
[24] H. Qu, K. Wang, and J. Zhao, "Survivable SFC deployment method based on federated learning in multi-domain network," *The Journal of Supercomputing*, vol. 79, no. 16, pp. 18198-18226, 2023.
[25] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," *Artificial Intelligence and Statistics PMLR*, pp. 1273-1282, 2017.
[26] Q. Xue, Y. Zhu, and J. Wang, "Mean estimation over numeric data with personalized local differential privacy," *Frontiers of Computer Science*, vol. 16, p. 163806, 2022.
[27] C. Garbin, X. Zhu, and O. Marques, "Dropout vs. batch normalization: An empirical study of their impact to deep learning," *Multimedia Tools and Applications*, vol. 79, no. 19, pp. 12777-12815, 2020.
[28] Y. Zhang, G. Wang, T. Yang, T. Pang, Z. He, and J. Lv, "Compression of deep neural networks: Bridging the gap between conventional-based pruning and evolutionary approach," *Neural Computing and Applications*, vol. 34, no. 19, pp. 16493-16514, 2022.

AUTHORS

Jingge Gao received her M.S. Degree from Hebei University. She is currently a lecture at Hebei University of Engineering. Her main research interests include detection methods, control theory, etc. She has published more six papers.

Shuqiang Zhang received his M.S. Degree from Xi'an Technological University. He is currently a lecture at Handan Polytechnic College. His main research interests include image processing, AI, control technique, etc. He has published more six papers.

Wei Lu received his M.S. Degree from Xi'an Technological University. He is currently a lecture at Hebei University of Engineering. Her main research interests include information collection, electrical control, etc.