Computer Science and Information Systems 20(1):1–24 https://doi.org/10.2298/CSIS220210051G

Homomorphic Encryption Based Privacy-Aware Intelligent Forwarding Mechanism for NDN-VANET

Xian Guo[1], Baobao Wang, Yongbo Jiang, Di Zhang, and Laicheng Cao

School of Computer and Communication,
Lanzhou University of Technology, Gansu 730050, China
iamxg@163.com
{1244737514, 670342320, 463923303, 28140795}@qq.com

Abstract. Machine learning has been widely used for intelligent forwarding strategy in Vehicular Ad-Hoc Networks (VANET). However, machine learning has serious security and privacy issues. BRFD is a smart Receiver Forwarding Decision solution based on Bayesian theory for Named Data Vehicular Ad-Hoc Networks (NDN-VANET). In BRFD, every vehicle that received an interest packet is required to make a forwarding decision according to the collected network status information. And then decides whether it will forward the received interest packet or not. Therefore, the privacy information of a vehicle can be revealed to other vehicles during information exchange of the network status. In this paper, a Privacy-Aware intelligent forwarding solution PABRFD is proposed by integrating Homomorphic Encryption (HE) into the improved BRFD. In PABRFD, a secure Bayesian classifier is used to resolve the security and privacy issues of information exchanged among vehicle nodes. We informally prove that this new scheme can satisfy security requirements and we implement our solution based on HE standard libraries CKKS and BFV. The experimental results show that PABRFD can satisfy our expected performance requirements.

Keywords: VANET, Bayesian decision theory, BRFD, homomorphic encryption.

1. Introduction

VANET has the characteristics of high-speed movement of vehicle nodes and frequent changes of network topology, which will cause frequent disconnection of the link between vehicle nodes [1, 2]. To ensure the reliability and stability of network connection, the routing protocol is a key to affect the performance of the VANET. In recent years, many intelligent solutions based on machine learning have been proposed [3-9]. Literature [10] made a comprehensive research on the security and management challenges for applying machine learning in VANET. To improve the VANET performance, the Bayesian classification algorithm has been widely used in predicting vehicle behavior [11-15]. In [16], we proposed a Bayesian-based Receiver Forwarding

[1] Corresponding author

Decision (BRFD) scheme to solve the broadcast storm in Named Data Vehicular Ad Hoc Networks (NDN-VANET).

Although machine learning has been widely used for VANET and other environments [17], the security and privacy problems in machine learning have been a focus of academia and business. A comprehensive investigation of privacy and security issues in machine learning is made in the literature [18]. Machine learning consists of two stages: the training stage and the testing stage. The poisoning attack[19, 20] is the best-known attack method in the training stage, and the attacks aiming at the testing stage include the membership inference attack[21], the evasion attack [22], and the model extraction attack[23, 24]. In this paper, we study security and privacy issues in the testing stage of BRFD.

The privacy-preserving machine learning (PPML) is firstly proposed by Lindell et al. [25]. The PPML allows two participants to extract the joint dataset without revealing their privacy. The early researches of PPML mostly used Yao's garbled circuit protocol [26], which has a large computational and communicating overhead. At present, the privacy protection technologies to achieve PPML have three broad categories, which are based on differential privacy (DP) [27-31], secure multi-party computation (SMPC) [32-38], and homomorphic encryption (HE) [39-44]. Adding noise to the sensitive data is a key method of differential privacy to achieve privacy protection. However, the increasing noise may lead to an accuracy decrease of machine learning. The schemes based on secure multi-party computation require multiple information interactions between participants, which is not suitable for the NDN-VANET with fast topology changes. Homomorphic encryption supports calculation on ciphertext, which can ensure the classification accuracy of the Bayesian model. Therefore, this paper adopts homomorphic encryption mechanism to solve security and privacy issues in BRFD we proposed in [16].

In BRFD, the vehicles exchange network status information in plaintext to make forwarding decision and no cryptographic mechanism is used in BRFD. The vehicle's privacy information such as location, speed, and so forth can be revealed to other vehicles. Aiming at the security and privacy issues caused by information exchange of network status used by machine learning in BRFD, a Privacy-Aware intelligent forwarding solution PABRFD is proposed by integrating HE [45] into BRFD in this paper. In PABRFD, HE is integrated into the Bayesian-based forwarding decision to protect the network status information of the vehicle. In addition, we improve the calculation method of the Bayesian probability values in BRFD and enhance the efficiency of the classification protocol by removing the complicated Gaussian formula calculation. We implement our novel scheme based on the CKKS library and BFV library [46] and make a performance comparison. We also informally analyze the security attributes of the PABRFD.

The remainder of this paper is structured as follows. The related works are reviewed in Section 2. The theoretical knowledges related to PABRFD are introduced in Section 3. The detailed PABRFD is described in Section 4. The experimental results of the PABRFD scheme are analyzed in Section 5. Finally, the conclusion and future work are introduced in Section 6.

2. Related Works

At present, researchers have proposed many privacy-preserving solutions based on HE aiming at machine learning for various applications. In this section, we review some solutions for VANET. In the vehicle-to-everything (V2X) communication system, in order to realize intelligent communication, vehicles and infrastructure equipment need to exchange data regularly. Therefore, the confidentiality and integrity of data need to be protected in an unverified and untrusted environment. Ulybyshev et al. [39] proposes a HE-based secure data exchange mechanism to protect the communication privacy between vehicles. The solution provides an access control scheme based on roles and attributes, which can detect and prevent data leakage caused by internal users. In addition, the authors propose a search method based on HE, which can query a vehicle's record stored on an untrusted cloud server based on ciphertext. The authors prove that their solution can protect vehicle and its owner's sensitive information against curiosity or malicious attacks.

Kong et al. [40] proposes a HE-based VANET secure data sharing scheme to protect a vehicle's private data. Each vehicle node is required to build a comprehensive data report and send the data report to RSU for secure data aggregation. Finally, the aggregated results will are sent and stored in a traffic management agency. After receiving a data query request, the RSU will share the aggregated result with the vehicle node.

In VANET, a reputation system often is used to judge whether a vehicle agrees to communicate with the target vehicle or not, according to the feedback information of other vehicles. So the feedback information plays a crucial role in the trust evaluation of neighbor nodes. In [41], a privacy-preserving vehicle feedback (PPVF) scheme is proposed based on HE and the data aggregation technique for VANET with cloud assistant system. The cloud service provider obtains the parameters related to vehicle in the vehicle feedback information, which is used for reputation calculation without revealing the private information of the vehicle that provides the feed information. Theoretical analysis and simulation experiment show that PPVF can achieve privacy protection for the feedback vehicle and PPVF has acceptable computational accuracy and communication consumption.

As machine learning-based routing algorithm is widely used in VANET, routing scheme also faces various security threats. An opportunistic routing protocol (ePRIVO) for vehicular delay-tolerant networks (VDTN) based on HE is proposed in [42]. The ePRIVO can protect some sensitive information during a routing decision of a vehicle. The ePRIVO models VDTN as a time-varying neighboring graph, and the graph's edge corresponds to the neighboring relationship between vehicles. In the ePRIVO, vehicles use HE to calculate the graph's similarity and secretly compare route metrics. Furthermore, their experimental results and analysis show that the accuracy of the ePRIVO is about 29% higher than other related routing protocols for privacy-preserving.

Alamer et al. [43] propose a privacy-preserving bidding framework VCC for VANET based on HE to protect the private interaction between a vehicle and a cloud server. An incentive mechanism is used in the bidding framework to encourage the interaction between the cloud server and the vehicle. The cloud server selects a participation vehicle to complete a task in cooperation. The selected vehicle will receive a certain

reward after the task is completed. Moreover, this mechanism ensures the authenticity of all participants and provides an allocation rule that enables the VCC framework to select the best resources for the task. In addition, due to using the HE technology, VCC and RSU can run an effective bidding process without acquiring the sensitive information of a vehicle.

A decentralized privacy-preserving deep learning model (DPDL) is proposed by integrating deep learning, blockchain, and FHE into VANET in [44]. DPDL can effectively reduce network communication overhead and congestion delay by decomposing computing tasks from a centralized cloud service to edge computing (EC) nodes. Blockchain is used to establish a secure and reliable data communication mechanism between RSU and EC nodes. In addition, the DPDL model provides a privacy-preserving data analysis scheme for VANET, and the fully homomorphic encryption (FHE) is used to encrypt the traffic data on each EC node and input it to the local DPDL model, thereby it can effectively protect the privacy and trustworthiness of the vehicle. Using of Blockchain can provide a reliable distributed update mechanism for the DPDL model, and the parameters of each local DPDL model are stored in the blockchain to share with other distributed models. In this solution, all distributed models can update their models in a reliable and asynchronous manner.

3. Preliminaries

3.1. Bayesian Theory Foundation

Bayesian Classification Algorithm

Bayesian classification algorithm [47], which is widely used for sample classification, is based on the Bayes theorem. The Naive Bayesian classification algorithm is one of the Bayesian classification algorithms. The Naive Bayes classification algorithm assumes that each attribute value is independent with the others and does not affect the classification results. The idea of the Naive Bayes classification algorithm is shown as follows.

Let $X = \{x_1, x_2, \cdots, x_m\}$ is an item to be classified, each $x_i (i = 1,2,\cdots, m)$ is a feature value of X. Given a set of categories $Y = \{y_1, y_2, \cdots, y_n\}$ where each $y_i (i = 1,2,\cdots, n)$ represents a category. If the posterior probability $P(y_k|X) = max\{P(y_1|X), P(y_2|X), \cdots, P(y_n|X)\} P(y_i|X) (i = 1,2,\cdots, n)$, X belongs to the *k*-th classification.

The posterior probability of each category $P(y_i|X)$ is shown as follows.

$$P(y_i|X) = \frac{P(X|y_i)P(y_i)}{P(X)}. \quad (1)$$

$P(X|y_i)$ is the prior probability of X, $P(y_i)$ is the probability of each category.

Bayesian Classifier

The Bayesian classifier [48] is a simple probabilistic classifier based on a Naive Bayesian algorithm. In this classifier, the model w consists of several probabilities: $\{P(y_i)\}_{i=1}^{n}$ is the probability of each category y_i, and $\{\{P(x_j|y_i)\}_{i=1}^{k}\}_{j=1}^{m}$ is the prior probability (When the X belongs to the i-th category y_i, the j-th feature value of X is x_j. m is the dimension of the X, k is the total number of categories). The classifier selects the category of the highest posterior probability as the final decision result, and the decision result is denoted as k_0.

$$\begin{aligned} k_0 &= P(y_i|X) \\ &= \underset{i\in[k]}{argmax} \frac{P(X|y_i)P(y_i)}{P(X)} \\ &= \underset{i\in[k]}{argmax} P(X|y_i)P(y_i). \end{aligned} \tag{2}$$

In the above formulation, removing the denominator $P(X)$ does not affect the final result due to the characteristics of the *argmax* function. In the Naive Bayes classifier, the m feature values of X are independent of each other, k_0 is shown as follows.

$$k_0 = \underset{i\in[k]}{argmax} P(y_i) \prod_{j=1}^{m} P(x_j|y_i) . \tag{3}$$

3.2. BRFD

Due to the characteristics of faster computing speed and higher classification accuracy, the Bayesian classification algorithm is widely used in VANET to improve network performance by predicting vehicle's behavior. In [16], we proposed a scheme called BRFD based on the Naive Bayes classifier to mitigate the broadcast storm problem incurred by interest packets in NDN-VANET. The BRFD mainly consists of three stages: the HELLO interaction, the Naive Bayesian decision, and the back-off forwarding.

HELLO Interaction

In BRFD, the special interest packets with the HELLO tag are used to regularly exchange the network status information $C = \{(x_i, y_i), speed_i, dis_i, num_i, D_i\}$ between neighboring nodes. In C, (x_i, y_i) denotes the vehicle location, $speed_i$ denotes the vehicle speed, dis_i denotes the distance, num_i denotes the number of neighbor vehicles, and D_i denotes the Bayesian Decision result. When the neighbor vehicle receives the HELLO packet, it will store the vehicle status information in its Decision Neighbor List (DNL). DNL is shown as follows.

$$DNL = \begin{pmatrix} x_1 & y_1 & speed_1 & dis_1 & num_1 & D_1 \\ x_2 & y_2 & speed_2 & dis_2 & num_2 & D_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_n & y_n & speed_n & dis_n & num_n & D_n \end{pmatrix} \quad (4)$$

Naive Bayesian Decision

In BRFD, after a vehicle *a* receives an interest packet from the neighboring vehicle *s*, firstly, the vehicle *a* reads the network status information $C = \{(x_i, y_i), speed_i, dis_i, num_i, D_i\}$ of the vehicle *s* to obtain the Bayesian decision condition $\{dis(s,a), speed_a, dis_a, num_a\}$, and then calculates the forwarding probability of $P(F|C)$ and the non-forwarding probability $P(\bar{F}|C)$ of the received interest packet according to the DNL. Finally, the vehicle decides whether it will enter the back-off forwarding process or not by comparing the value of the $P(F|C)$ and the $P(\bar{F}|C)$.

Back-off Forwarding

The BRFD is a scheme based on the receiver-forwarding decision. Each vehicle does not know whether other vehicles also will forward the received interest packet. Therefore, a conflict may be incurred by the same copies of an interest packet due to multiple vehicle nodes forward the same interest packet. To solve this problem, a back-off forwarding mechanism is adopted. Each node will set a back-off delay according to its forwarding probability calculated in the Naive Bayesian decision stage.

3.3. Homomorphic Encryption

HE[49] is a cryptographic technique based on a certain mathematical problem. The detailed description of HE is as follows. Let p_1 and p_2 be two plaintexts in the plaintext space M, *PK* is the public key and *SK* is the private key respectively. *Enc* and *Dec* are two algorithms for encryption and decryption respectively. "⊙" represents operation on ciphertexts, "·" represents operation on plaintexts. The HE operation satisfies the following equation.

$$p_1 \cdot p_2 = Dec_{SK}(Enc_{PK}(p_1) \odot Enc_{PK}(p_2)) . \quad (5)$$

HE is one of the common privacy protection technologies. Due to the computation on the two ciphertexts is equal to the direct computation on the two plaintexts as shown in equation (5), it is widely used for privacy-preserving in machine learning applications. At present, HE can be divided into Partially Homomorphic Encryption (PHE), Somewhat Homomorphic Encryption (SHE), and Fully Homomorphic Encryption (FHE). PHE only allows one type of operation but does not limit the number of operations. The classic cases of PHE are EI-Gamal[50], Paillier[51] and RSA[52]. SHE allows multiple types of operations, but the number of operations is limited. The classic

case of SHE is BGN[53]. FHE does not limit the type and number of operations. The classic case of FHE is Gentry[45].

The SEAL library provides two fully homomorphic encryption schemes: the BFV scheme and the CKKS scheme [54]. The BFV scheme supports integers operation, and the CKKS scheme supports floating-point numbers operation. There are a lot of floating-point numbers in the process of Bayesian classification probability calculation. So, when the BFV scheme is used, it is necessary to convert the eigenvalues in the motion data to integers by an expansion factor. This process will lead to a loss of data accuracy, further reducing the model classification accuracy. So, we adopt the CKKS scheme to implement our PABRFD scheme in this paper.

CKKS allows the addition and multiplication of encrypted real or complex numbers. It is a common scheme for privacy-preserving in machine learning applications. CKKS includes six basic algorithms: Key generation algorithm (*KeyGen*), Encode algorithm (*Ecd*), Decode algorithm (*Dcd*), Encryption algorithm (*Enc*), Decryption algorithm (*Dec*), and ciphertext calculation algorithm (*Eval*).

The CKKS is designed based on RLWE. The plaintext space of RLWE is a polynomial ring $\mathbb{R} = \mathbb{Z}[X]/\Phi_M(X)$ (Here, $\Phi_M(X)$ is the M-th cyclotomic polynomial of degree $N = \Phi(M)$, M is a positive integer). However, the plaintext space of CKKS is a complex vector space $\mathbb{C}^{\Phi(M)/2}$. Therefore, it is necessary to find the mapping relationship between the two. CKKS defines the canonical embedding mapping σ: $\mathbb{R} \to \mathbb{H}$ and the natural projection π: $\mathbb{H} \to \mathbb{C}^{\Phi(M)/2}$, both π and σ are full mappings, and their inverse mappings are $\pi^{-1} and\ \sigma^{-1}$. Let $\mathbb{H} = \left\{(z_j)_{j \in \mathbb{Z}_M^*} : z_{-j} = \overline{z_j}, \forall j \in \mathbb{Z}_M^*\right\} \subseteq \mathbb{C}^{\Phi(M)}$, Let T be a multiplicative subgroup of $\mathbb{Z}_M^*$ satisfying $\mathbb{Z}_M^*/T = \{\pm 1\}$. For a vector $z \in \mathbb{C}^{\Phi(M)/2}$, the encoding procedure first expands it into the vector $\pi^{-1}(z) \in \mathbb{H}$, and then computes its discretization to $\sigma(\mathbb{R})$ after multiplying a scaling factor Δ. At last, the corresponding integral polynomial is $m(X) = \lfloor \Delta \cdot \pi^{-1}(z) \rceil_{\sigma(\mathbb{R})}$. The decoding procedure of CKKS is the inverse process of the encoding algorithm, so it will not be introduced in detail.

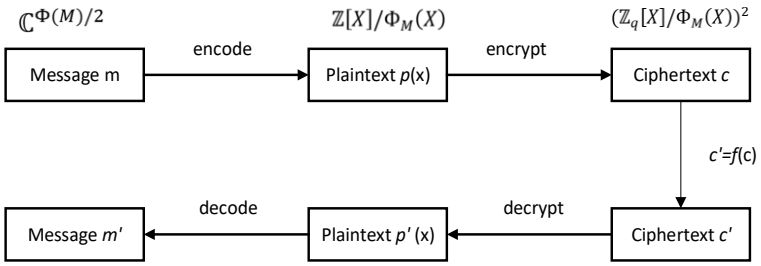

Fig.1. Overview of CKKS Scheme

An overview of the CKKS scheme is shown as follows Fig.1: A message m that used to perform a specific computation is first encoded into plaintext polynomial $p(X)=Ecd(m)$ and then is encrypted by using the public key. Once a message m is encrypted into a ciphertext $c=Enc(p(X))$, the CKKS scheme provides addition and multiplication operations. A combination of homomorphic operations is denoted as f. Decrypting c' with the secret key will obtain $p'(X)=Dec(c')$ and then decoding to $m'=Dcd(p'(X))$. More detailed techniques are mentioned in [54].

3.4. Argmax Protocol

The *argmax* function is usually used to obtain the subscript of the maximum value in the set. In [55], the authors combine the *argmax* protocol with HE to obtain the subscript of the maximum value in the Bayesian classification results. Assuming a participant *A* holds a set of encrypted probability values encrypted with the counterpart's public key *pk*, and the other participant *B* has a decryption key *sk*. For convenience, $[[x]]$ denotes $Enc_{pk}(x)$. Let $[[a]]$ and $[[b]]$ denote two ciphertexts to be compared. Let $\mathcal{G}$ denotes a set of a linear polynomial from $g(x)=Cx$ where *C* is a positive integer. Since $g \in \mathcal{G}$ is a linear polynomial with positive coefficients, we obtain $g(a) - g(b) \geq 0$ when $a \geq b$ and $g(a)-g(b) < 0$ otherwise.

When comparing two ciphertext values, *A* randomly chooses $g \in \mathcal{G}$ and computes $[\![h]\!]=[\![g(a)]\!] \ominus [\![g(b)]\!]=g([\![a]\!]) \ominus g([\![b]\!])$. Then *A* sends $[\![h]\!]$ to *B* who decrypts it to obtain *h*. if $h \geq 0$, *B* updates $index$ and generates a ciphertext $[\![d]\!]=[\![1]\!]$, which is an encryption of vector whose values are all 1. Otherwise, *B* will not update $index$ and generate $[\![d]\!]=[\![0]\!]$, which is an encryption of vector whose values are all 0. *B* sends $[\![d]\!]$ to *A*, who then computes $([[d]] \otimes [[a]]) \oplus ((1 \ominus [[d]]) \otimes [[b]])$. The resulting ciphertext is the higher value and will be used in the next comparison. The comparison is repeated until all values have been compared. Once all comparisons are made, *B* sends the *index* to *A*, who reverses the permutation to obtain the actual *index i*.

4. Our Solution PABRFD

In this section, we present our new solution PABRFD. To explain our PABRFD, we firstly describe a simplified NDN-VANET network model as shown in Fig. 2. In this network model, the vehicle nodes are roughly divided into two categories: The vehicle node *R* and the neighbor vehicle nodes of *R* (They are denoted by a set $S = \{S_1, S_2, \cdots, S_n\}$). Here, we assume that the vehicle node *R* received an interest packet from some adjacent vehicle node S_i and then it needs to decide whether it should forward the received interest packet or not according to our PABRFD solution. The neighboring vehicles of *R* in S will periodically provide network status information to *R* by a special HELLO packet and *R* will store these information in DNL according to BRFD [16]. The network status information is important data that is used to make an intelligent forward decision by the vehicle *R*.

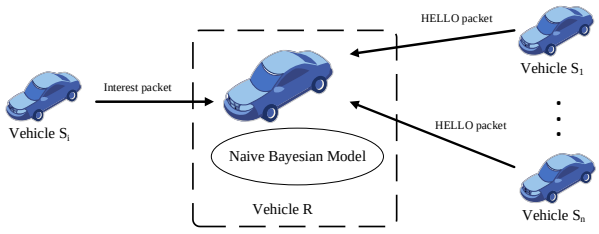

Fig.2. The Simplified Network Model

4.1. Security Requirements

To ensure secure and privacy exchange of network status information, we assume that the vehicle nodes in the set *S* are semi-honest, so the PABRFD scheme should satisfy the following security requirements:

1. **System correctness**: A vehicle node that received an interest packet can correctly make a forward decision.
2. **Data integrity**: It is required that the content in a HELLO packet cannot be modified when it is transmitted in NDN-VANET.
3. **Data confidentiality**: Some sensitive information such as the vehicle location and speed and so forth, will not be revealed when the HELLO packet is transmitted in NDN-VANET.

4.2. System Building Blocks

To achieve these security requirements mentioned in section 4.1. We propose a novel scheme PABRFD based on BRFD [16] by integrating security mechanisms such as HE into BRFD in this paper and by improving BRFD. This PABRFD consists of the following modular.

Key Generation

In PABRFD, all vehicle nodes need to initialize their own public key, private key, and evaluation key. We assume that λ is a security level parameter, and L represents the upper limit of the ciphertext length. Choose a large integer $p>0$ as the base and q_0 as the modulus, let $q_l = p^l \cdot q_0, 0 < l \leq L$, l is the ciphertext depth. For a real number $\sigma>0$, $Dg(\sigma^2)$ extracts a vector in $\mathbb{Z}^N$ by drawing its coefficient from the discrete Gaussian distribution of variance σ^2. For a real number $0\leq\rho\leq1$, the distribution $ZO(\rho)$ draws each entry in the vector from $\{0,-1,1\}^N$, where the probability of selecting 1 and -1 is $\rho/2$, and the probability of selecting 0 is $1-\rho$. For a positive integer h, $HWT(h)$ is the set signed N-dimensional binary vector set $\{0,1,-1\}^N$ whose Hamming weight is exactly h.

Key generation algorithm $Keygen(1^{\lambda})\text{->}(pk, sk, evk)$: Generate a secret key sk, a public key pk for encryption, and an evaluation key evk.

According to the security parameters λ and q_L, choose a power-of-two $M = M(\lambda, q_L)$ for cyclotomic polynomial, an integer $h = (\lambda, q_L)$, integer $P = P(\lambda, q_L)$, real number $\sigma = \sigma(\lambda, q_L)$.

$$s \leftarrow HWT(h), a \leftarrow \mathbb{R}_{q_L}, e \leftarrow Dg(\sigma^2). \tag{6}$$

Let the private key $sk \leftarrow (1, s)$, the public key $pk \leftarrow (b, a) \in \mathbb{R}_{q_L}^2$, where $b \leftarrow -a \cdot s + e \ (mod\ q_L)$, $\mathbb{R}_{q_L}$ represents a polynomial ring. Sample $a' \leftarrow \mathbb{R}_{p \cdot q_L}, e' \leftarrow Dg(\sigma^2)$, and then the evaluation key evk is

$$evk \leftarrow (b', a') \in \mathbb{R}_{P \cdot q_L}^2. \tag{7}$$

where $b' \leftarrow -a' \cdot s + e' + P \cdot s^2 (mod\ P \cdot q_L)$.

Bayesian Model Construction

The Bayesian model consists of the prior probability of each feature attribute and the class probability of each category. We assume that $P(F_i)(i = 1,2)$ denotes Interest Forwarding Event, F_1 means that the vehicle node R will forward the received interest packet, and F_2 means that the vehicle node R won't forward the interest packet.

Let N_{F_1} be the number of interest packets forwarded in the training set. Let N_{F_2} be the number of interest packets not forwarded in the training set, then the forwarding and non-forwarding probabilities are shown as follows.

$$P(F_i) = \frac{N_{F_i}}{N_{F_1} + N_{F_2}} \ (i = 1,2) \ . \tag{8}$$

Let the $N_{(dis^2|F_i)}$ is the number of the *dis* in the training set that the decision result is $F_i(i = 1,2)$. Let the $N_{(num|F_i)}$ is the number of the *num* in the training set that the decision result is $F_i(i = 1,2)$. Let the $N_{(speed|F_i)}$ is the number of the *speed* in the training set that the decision result is $F_i(i = 1,2)$. Then the prior probability of each feature value is shown as follows.

$$\begin{aligned} P(dis^2|F_i) &= \frac{N_{(dis^2|F_i)}}{N_{F_i}} \ . \\ P(\text{num}|F_i) &= \frac{N(num|F_i)}{N_{F_i}} \ . \\ P(speed|F_i) &= \frac{N(speed|F_i)}{N_{F_i}} \ . \end{aligned} \tag{9}$$

The vehicle node R stores the probability of classification $F_i(i = 1,2)$ and the prior probability of each feature value into a vector of length t+1(t is the number of feature values, t=3 in this scheme), $W_{F_i}[j] = P(C_j|F_i)$,$W_{F_i}[t+1] = P(F_i)(i = 1,2,j = 1,2,3)$, $C = \{dis^2, speed, num\}$, C_j represents the j-th element in C. In model $W = \{w_1, w_2\}$, w_1 and w_2 represent forwarding and non-forwarding categories respectively, and the model matrix W is shown as follows.

$$Model\ W = \begin{bmatrix} P(C_1|F_1) & P(C_2|F_1) & P(C_3|F_1) & P(F_1) \\ P(C_2|F_2) & P(C_2|F_2) & P(C_3|F_2) & P(F_2) \end{bmatrix} \tag{10}$$

4.3. Implement of PABRFD

In this section, we present the implementation details of the PABRFD scheme. PABRFD can be divided into the following four stages: System Initialization stage, Network Status Information Exchange stage, Naive Bayes Decision stage, and Secure Argmax. A detail flow chart of the PABRFD is described in figure 3 :

System Initialization

In this system initialization process, we firstly initialize cryptographic materials as described in section 4.2.1. And then we will prepare the Bayesian model for PABRFD according to section 4.2.2. In our novel scheme, we use SUMO traffic simulation software [56] to generate vehicle motion data. The simulation area of the network road is set to 200m*200m. The number of vehicle nodes in the network is 10-60. The speed of the vehicle is between 0-100 mph. In BRFD [16], *dis*, *speed*, and *num* are regarded as continuous values and assume that these continuous values obey a Gaussian distribution of mean μ and variance σ^2. The conditional probability of each feature value is calculated by the following formula.

$$g(x,\mu,\sigma) = \frac{1}{\sqrt{2\Pi}\sigma} e^{-(\frac{(x-\mu)^2}{2\sigma^2})}. \quad (11)$$

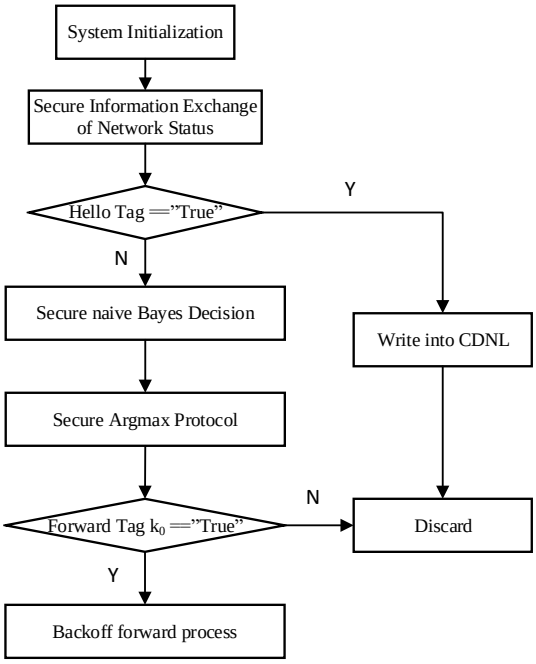

Fig.3. PABRFD flow chart

In PABRFD, to improve the computational efficiency of HE, we choose to discretize *dis*, *speed*, and *num*. The value of *dis* is between 0 and $200\sqrt{2}$. In our simulation experiment, we use the square of *dis* as a feature value to train the model. Therefore, the value of dis^2 is between 0 and 80000, we divide it into 400 intervals, and replace the entire interval with the middle value of each interval. The set of dis^2 after division is $\{100,300,\cdots,79900\}$. In the same way, the value of *speed* is between 0 and 100, we divide it into 10 intervals, and the set of *speed* after division is $\{5,15,\cdots,95\}$. The *num* is between 0-60, we divide it into 10 intervals, and the set of *num* after division is

$\{0,3,9,15,\cdots,57\}$ (When the number of neighboring nodes is 0, the node cannot forward interest packets, so 0 is a special value and needs to be divided separately).

The algorithm of the Bayesian model training stage is shown in Algorithm 1:

Secure Information Exchange of Network Status

In PABRFD, all nodes in the network need to periodically send their network status information to their neighbor nodes. We assume that some nodes $S_i(i = 1,2,\cdots)$ need to send its network status information. It will firstly encode the network status information $C = \{(x_i, y_i), speed_i, dis_i, num_i, D_i\}$ into polynomial $m(C)$ and then encrypts $m(C)$, the encryption process does as follows:

Sample $v \leftarrow ZO(0.5)$, $e_0, e_1 = Dg(\sigma^2)$.

Final output:

$$c \leftarrow v \cdot pk_{S_i} + (m(C) + e_0, e_1)(mod\ q_l)\ . \tag{12}$$

And then the node S_i writes c into an interest packet with a HELLO tag and broadcasts the interest packet to its neighboring nodes. The vehicle node that received the HELLO packet will update Ciphertext Decision Neighbor List (CDNL) according to the information carried in this HELLO packet.

Algorithm 1: Bayesian model training algorithm

Input: Vehicle motion datasets $C = \{(x_i, y_i), speed_i, dis_i, num_i, D_i\}$
Output: Model *W*
//Data discretization
$dis^2 : (0,80000) \rightarrow \{100,300,\cdots,79900\}$
$speed : [0,100] \rightarrow \{5,15,\cdots,95\}$
$num : [0,60] \rightarrow \{0,3,9,15,\cdots,57\}$
//Calculate the class probability and the prior probability of each feature attribute
for *i*=1 to 2:
$P(F_i) = \frac{N_{F_i}}{N_{F_1} + N_{F_2}}$. //Calculated the class probability
$P(dis^2|F_i) = \frac{N_{(dis^2|F_i)}}{N_{F_i}}$.
$P(\text{num}|F_i) = \frac{N(num|F_i)}{N_{F_i}}$. // Calculate the prior probability
$P(speed|F_i) = \frac{N(speed|F_i)}{N_{F_i}}$.
end for
//Build the model matrix *W*
for *i*=1 to 2:
for *j* = 1 to 3:
$W_{F_i}[j] = P(C_j|F_i)$ // Write the prior probability
end for
$W_{F_i}[j] = P(F_i)$ // Write the class probability
end for
Return model *W*

Secure Bayesian Decision

We assume that a vehicle node *R* receives an interest packet from some vehicle node $S_i(i = 1,2,\cdots)$that want to request some interested data on the network, the vehicle node *R* will extract the encrypted network status information $\{([[x_s]],[[y_s]]),[[speed_s]],[[num_s]],[[D_s]]\}$ from CDNL. And then the node *R* will use homomorphic encryption to calculate the values used in the secure Bayesian Decision. Firstly, it calculates the square of the distance [[dis^2(*S*,*R*)]], and then constructs the item $[[C]] = \{[[dis^2(S,R)]], speed_R, num_R\}$ to be classified. For each class F_i(*i*=1,2), the node *R* matches [[*C*]] with the Naive Bayesian model *W*={w_1, w_2} and calculates the posterior probability [[$P(F_i|C)$]].The [[dis^2(*S*,*R*)]] is shown as follows.

$$\begin{aligned}[[dis^2(S,R)]] &= ([[x_S]] - x_R)^2 + ([[y_S]] - y_R)^2 \\ &= [[x_S]]^2 + [[y_S]]^2 - 2 \cdot x_R \cdot [[x_S]] - 2 \cdot y_R \cdot [[y_S]] + x_R^2 + y_R^2 .\end{aligned} \quad (13)$$

The posterior probability $P(F_i|C)$ can be calculated by the following formula:

$$[[P(F_i|C)]] = \frac{[[P(C|F_i)]] \cdot P(F_i)}{P(C)} (i = 1,2) . \quad (14)$$

In the above formula, $P(F_i)$ is the class probability, which has been obtained in the Bayesian training stage. Due to the characteristics of the *argmax* protocol (Details are shown in 3.1.2), $P(C)$ does not affect the final result, so it is not necessary to calculate. We assume that the feature values of the network status information are independent, so $P(C|F_i)$ can be calculated by the following formula:

$$[[P(C|F_i)]] = [[P(([[dis^2(S,R))]]|F_i)]]P(speed_R|F_i)P(num_R|F_i) . \quad (15)$$

The three conditional probabilities can be calculated by the formula (9) introduced in the Bayesian model training stage. Bring the calculation result of the formula (15) into the formula (14) to obtain the posterior probability [[$P(F_i|C)$]].

The detailed algorithm is shown as follows.

Algorithm 2: Secure naive Bayes Decision

Input: Items to be classified $[[C]] = \{[[dis^2(S,R)]], speed_R, num_R\}$, natural projection π, Naive Bayesian model *W*={w_1, w_2}

Output: The ciphertext sequence $\{[[P_{\pi(i)}]]\}$

//calculates the forwarding and non-forwarding probability [[$P(F_i|$C)]](i=1,2)

for *i*=1 to 2 do

temp ←[[C]] ⊗*W*, //Match every probability

[[$P(F_i|C)$]]←add(*temp*)

end for

The ciphertext sequence is

$$\{[[P_{\pi(i)}]]\} = [[P(Fi|C)]] = P(F_i)\prod_{j=1}^{t}[[P(C_j|F_i)]]\}_{i\in[1,2],j\in[1,t]}$$

Secure Argmax Protocol

After calculating the classification probability $\{[[P_{\pi(i)}]]\}(i = 1,2)$, the vehicle node R needs to interact with its neighboring vehicle nodes in S to obtain the final classification result k_0. That is to say, we need to compute the maximum value in these classification probabilities $\{[[P_{\pi(i)}]]\}(i = 1,2)$.

The *argmax* protocol is executed between the vehicle node S_i and the vehicle R, and the final forwarding decision result k_0 is output.

As mentioned in sections before, the interaction process between the vehicle node R and some neighboring node S_i is shown in figure 4.

The detailed algorithm is shown as follows.

Algorithm 3: Secure argmax protocol

R's input: Encrypted category probability $\{[[P_{\pi(i)}]]\}_{i\in[1,2]}$, Polynomial set $\mathcal{G}$

S's input: private key SK_S, and public key PK_S

Output: Subscript of maximum probability k_0

S:

index ←1

R:

$[[max]] \leftarrow [[P_{\pi(1)}]]$

for *i*=1 to 2 do

R:

Random sampling $g \in \mathcal{G}$

$[[temp]] \leftarrow g([[P_{\pi(i)}]]) \ominus g([[max]])$

send [[*temp*]] to *S*

S:

$temp \leftarrow Dec_{SK_S}([[temp]])$

if *temp*≥0 : *r*←1,*index*←*i*

if *temp*<0: *r*←0

$[[r]] \leftarrow Enc_{PK_S}(r)$

send [[*r*]] to *R*

R:

$[[max]] \leftarrow ([[r]] \otimes [[P_{\pi(i)}]]) \oplus ((1 \ominus [[r]]) \otimes [[max]])$

end for

S:

send *index* to *R*

R:

$k_0 = \pi^{-1}(index)$

Output category F_{k_0} is the final classification result

5. Security and Performance Analysis

5.1. Security Analysis

We informally prove that PABRFD can satisfy the security requirements proposed in 4.1.

Proposition 1. System correctness: Any vehicle node that received an interest packet can correctly make the forward decision.

***Proof*:** We assume that vehicle nodes are semi-honest in PABRFD, that is to say, a node will also follow the requirements of a protocol even if it is controlled by an attacker. In addition, PABRFD only introduces cryptographic mechanism to protect privacy in contrast with BRFD. So, the correctness of BRFD can ensure the correctness of PABRFD.

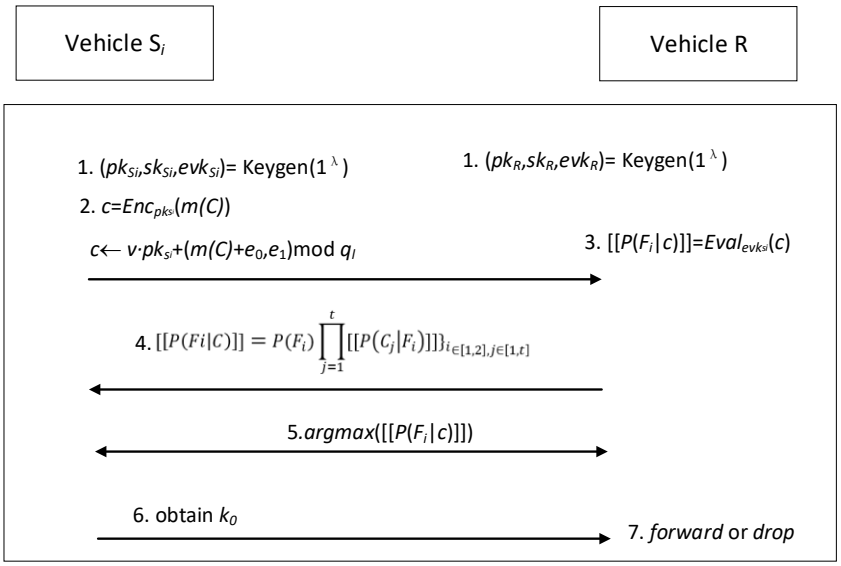

Fig.4. The interaction process between S_i and R

Proposition 2. Data integrity: Any malicious vehicle node can't modify data in the HELLO packet when it is transmitted in NDN-VANET.

***Proof*:** In PABRFD, HE is used to protect privacy in a HELLO packet when a vehicle node exchanges network status information with other vehicles. That is to say, a vehicle node will encrypt network status information such as location, speed, and so forth in the HELLO packet. Therefore, HE actually is a public key encryption scheme. So, using of HE in PABRFD can guarantee data integrity in the HELLO packet. Of course, data in the HELLO packet can't be modified by other nodes that they are malicious in semi-honest.

Proposition 3. Data confidentiality: Some sensitive information such as the vehicle location and speed and so forth, will not be revealed when the HELLO packet is transmitted in NDN-VANET.

***Proof*:** the proof is similar to proposition 2. The information in the HELLO packet is encrypted by using HE when it is transmitted in NDN-VANET or stored in CNDL. Therefore, the vehicle node without the corresponding decryption key cannot obtain the sensitive information of other vehicle nodes.

5.2. Performance Analysis

In this section, we evaluate PABRFD and two relative schemes such as BRFD and BFV-BRFD in terms of computational and communication efficiency. The BFV-BRFD that we specially designed to compare the performance with PABRFD is a scheme based on BRFD.

Experimental data and environment

The experimental data comes from SUMO [56] traffic simulation software. We use the ndnSIM [57] platform to evaluate and analyze our scheme PABRFD and the relative schemes. The simulation area of the network road is set to 200m*200m. The number of the vehicle nodes is 10-60, and the speed of the vehicle is set between 0-100 mph. The duration of each experiment is set to 100s, and the experiment results are average of 10 experiments we do.

Classification performance

As shown in the Table 1 below, we conduct experimental tests from different dimensions to evaluate the performances of BRFD, PABRFD, and BFV-BRFD schemes. We use three common metrics such as *Precision*, *Recall*, and *Accuracy* as follows.

$$\begin{aligned} Precision &= \frac{TP}{TP+FP}. \\ Recall &= \frac{TP}{TP+FN}. \\ Precision &= \frac{TP+TN}{TP+TN+FP+FN}. \end{aligned} \tag{16}$$

Here, parameters *TP*, *TN*, *FP*, and *FN* are explained as follows.

TP(*True Positive*): The true class of the sample is positive, and the predicted result is positive.

TN(*True Negative*): The true class of the sample is negative, and the predicted result is negative.

FP(*False Positive*): The true class of the sample is negative, and the predicted result is positive.

FN(*False Negative*): The true class of the sample is positive and the predicted result is negative.

Our experimental results in Table 1 show that the *Precision*, *Recall*, and *Accuracy* of the PABRFD and BFV-BRFD schemes are lower than those of the BRFD scheme without cryptographic mechanism. Integrating the encryption mechanism into BRFD in the PABRFD and BFV-BRFD schemes incurs the decline of the classification performance. In addition, in the PABRFD and BFV-BRFD schemes, the motion data is discretized to improve the computational efficiency of Bayesian classification probability. However, the discretized motion data also results in a decrease in the classification performance. It is worth mentioned that the classification performance of the PABRFD and BFV-BRFD schemes is still within an acceptable range. In addition, the classification performance of the PABRFD scheme is higher than that of the BFV-BRFD scheme, because the motion data needs to be converted into integers in the BFV-BRFD scheme. So, loss of the accuracy of the motion data also results in a decrease in the classification performance.

Table 1. Classification performance

scheme	*Accuracy*	*Precision*	*Recall*
BRFD	0.97	0.98	0.99
PABRFD	0.96	0.97	0.99
BFV-BRFD	0.93	0.94	0.98

Table 2. Time consumption comparison of homomorphic encryption

scheme	poly_modulus_degree	encryption (ms)	decrypt (ms)	addition (ms)	multiplication (ms)
BFV-BRFD	2048	287	18	11	80
	4096	1423	78	23	421
	8192	4810	241	71	1213
PABRFD	2048	314	16	7	65
	4096	1513	72	22	208
	8192	4818	233	70	841

Communication overhead

The computational time of the HE algorithm is highly related to the degree of polynomial modulus (poly_modulus_degree) and the modulus of ciphertext (coeff_modulus). The larger the poly_modulus_degree is, the security higher of the scheme is, but the ciphertext complex also increases, which will decrease the computational efficiency of the homomorphic operation. Therefore, in our experiments, we test the classification performance by using different polynomial modulus degrees:

2048, 4096, and 8192. As we can see in the Table 2 below, the performance of the PABRFD scheme is better than that of BFV-BRFD, especially in terms of multiplication operation time. As the poly_modulus_degree increases, the time consumption of the PABRFD scheme increases slowly.

In addition, to further evaluate and compare the performance of PABRFD, BRFD, and BFV-BRFD, we compare the computational performance of the three schemes in performing a Bayesian decision. As shown in Table 3, due to the introduction of the homomorphic encryption mechanism, the time consumption in running a Bayesian decision in the PABRFD and BFV-BRFD significantly increases. At the same time, as the poly_modulus_degree grows, the time consumption also increases exponentially and the result is similar in the PABRFD and BFV-BRFD. At the same time, the model *Accuracy* of the BFV-BRFD scheme is lower than that of BRFD. The BFV-BRFD scheme needs to convert floating-point numbers to integers in the Bayesian decision, which loses the *Accuracy* of the motion data. So, it is difficult for the BFV-BRFD scheme to achieve the accuracy of BRFD.

Table 3. Time consumption comparison in a Bayesian decision

Scheme	poly_modulus_degree	Average Time(ms)	Whether to achieve the *Accuracy* of BRFD
BRFD	/	3.8	/
BFV-BRFD	4096	614	no
	8192	1411	no
PABRFD	4096	519	yes
	8192	1267	yes

Like the work [16], we also introduce the following metrics such as *IPSD*, *NFIP*, and *NSIP* to evaluate the PABRFD scheme: *IPSD* indicates an interest packet satisfaction delay. That is to say, it is a time interval from a node sends an interest packet to the node receives a data packet related to the sent interest packet. *IPSD* is an important indicator to evaluate the NDN performance. *NFIP* notes a total number of forwarding interest packets in a simulation period. And *NSIP* indicates a total number of satisfied interest packets in the simulation period. That is to say, all of the nodes that sent these interest packets got the intended content.

Firstly, we compare the *IPSD* of PABRFD, BFV-BRFD, and BRFD. In NDN-VANET, high-speed movement of vehicles results in intermittent wireless links between adjacent nodes. So, the vehicle nodes should forward the received interest packets with minimal delay. As shown in Figure 5, after integrating the homomorphic encryption mechanism, the *IPSD* of the PABRFD scheme and the BFV-BRFD scheme is much higher than that of the BRFD scheme.

Secondly, we compare the *NFIP* of PABRFD, BFV-BRFD, and BRFD schemes. In the NDN-VANET network, *NFIP* represents the total number of Interests forwarded during the simulation, which is an important indicator for evaluating broadcast suppression. The lower the *NFIP*, it has the fewer redundant interest packets, and the better network performance. The *BRFD* scheme adopts a receiver-forwarding decision scheme based on Bayesian to suppress broadcast storms. As shown in Figure 6, the *NIFP* in the PABRFD is 5%-10% higher than that in the BFV-BRFD. Because the

classification accuracy of the BAPRFD scheme is higher than that of the BFV-BRFD scheme, the *NFIP* of the PABRFD scheme is closer to that of the BRFD scheme.

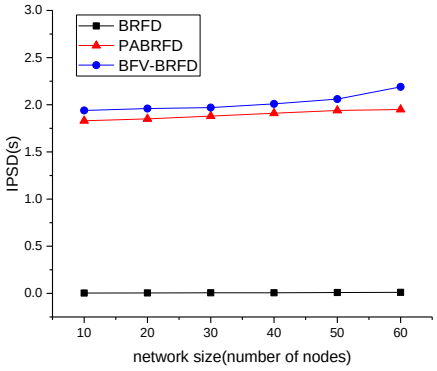

Fig. 5. IPSD

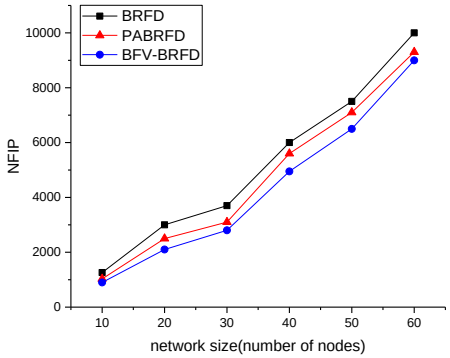

Fig. 6. NFIP

Finally, we analyze the *NSIP* of PABRFD, BFV-BRFD, and BRFD schemes during the simulation. *NSIP* represents the total number of Interest packets satisfied during the simulation, and *NSIP* also reflects the accuracy of the forwarding decision mechanism, which is an important evaluation index in NDN. As shown in Figure 7, the PABRFD and BFV-BRFD scheme reduces *NIPS*, mainly due to the decline of *NFIP*.

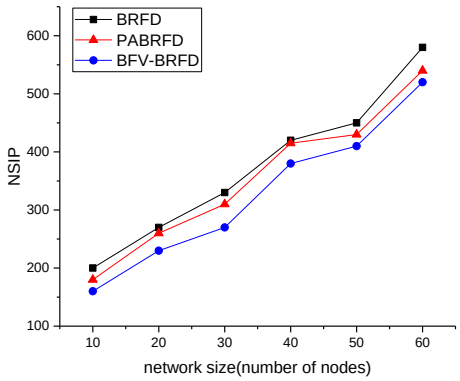

Fig. 7. NSIP

6. The conclusion and future work

In this paper, a Privacy-Aware intelligent forwarding solution PABRFD is proposed by integrating HE into BRFD. In PABRFD, a secure Bayesian classifier based on HE is used to protect the security and privacy of vehicle nodes. In PABRFD, First of all, the vehicle motion data in NDN-VANET is discretized to avoid the complex Gaussian calculation in the BRFD scheme, and the calculation efficiency of Bayesian classification probability is improved, which also provides a convenience for a secure and efficient naive Bayesian classifier. Secondly, we propose a HE-based secure network status exchange and storage mechanism suitable for the NDN-VANET environment, which can protect the private information of vehicles in NDN-VANET. Finally, we informally analysis security attributes that we hope to achieve. And we also implement the PABRFD and compare it's performance with related schemes BRFD and BFV-BRFD. Our experimental results show that our novel solution can satisfy our expected requirements.

However, the PABRFD scheme has limitations, and HE increases the time consumption of the PABRFD. Therefore, finding a solution with lower time consumption and better performance is one of our main research works in the future. At the same time, the security and privacy issues in the Bayesian training phase are also one of our future research works.

Acknowledgments: This work is supported by NSFC No. 61461027; Gansu province science and technology plan project under grant No. 20JR5RA467; Innovation Promotion Education Fund of Ministry of Education No. 2018A05003; Graduate Fine-designed Course of Lanzhou University of Technology.

References

1. M. S. Sheikh and J. Liang. A comprehensive survey on VANET security services in traffic management system. Wireless Communications and Mobile Computing. vol. 2019, (2019).
2. M. A. Hossain, R. M. Noor, K.-L. A. Yau, S. R. Azzuhri, M. R. Z'aba, and I. Ahmedy. Comprehensive survey of machine learning approaches in cognitive radio-based vehicular ad hoc networks. IEEE Access. vol. 8, 78054-78108. (2020).
3. L. Zhao, Y. Li, C. Meng, C. Gong, and X. Tang. A SVM based routing scheme in VANETs. in 2016 16th International Symposium on Communications and Information Technologies (ISCIT), IEEE, 380-383. (2016).
4. K. Roscher, T. Nitsche, and R. Knorr. Know thy neighbor-a data-driven approach to neighborhood estimation in vanets. in 2017 IEEE 86th Vehicular Technology Conference (VTC-Fall), IEEE, 1-5. (2017).
5. H. Bangui, M. Ge, and B. Buhnova. A hybrid machine learning model for intrusion detection in VANET. Computing. vol. 104, No. 3, 503-531. (2022).
6. R. Bibi, Y. Saeed, A. Zeb, T. M. Ghazal, T. Rahman, R. A. Said, et al. Edge AI-based automated detection and classification of road anomalies in VANET using deep learning. Computational intelligence and neuroscience. vol. 2021, (2021).
7. S. K. Singh, J. Cha, T. W. Kim, and J. H. Park. Machine learning based distributed big data analysis framework for next generation web in IoT. Computer Science and Information Systems. vol. 18, No. 2, 597-618. (2021).
8. C. Zhang, X. Zhao, M. Cai, D. Wang, and L. Cao. A new model for predicting the attributes of suspects. Computer Science and Information Systems. vol. 17, No. 3, 705-715. (2020).
9. N. Y. Yen, H.-Y. Jeong, K. Madani, and F. I. Massetto. Guest editorial: Emerging services in the next-generation web: Human meets artificial intelligence. Computer Science and Information Systems. vol. 18, No. 2, 1-6. (2021).
10. L. Liang, H. Ye, and G. Y. Li. Toward intelligent vehicular networks: A machine learning framework. IEEE Internet of Things Journal. vol. 6, No. 1, 124-135. (2018).
11. S. Ftaimi and T. Mazri. A comparative study of Machine learning algorithms for VANET networks. in Proceedings of the 3rd International Conference on Networking, Information Systems & Security, ACM, 1-8. (2020).
12. S. Khatri, H. Vachhani, S. Shah, J. Bhatia, M. Chaturvedi, S. Tanwar, et al. Machine learning models and techniques for VANET based traffic management: Implementation issues and challenges. Peer-to-Peer Networking and Applications. vol. 14, No. 3, 1778-1805. (2021).
13. T. Liu, S. Shi, and X. Gu. Naive Bayes Classifier Based Driving Habit Prediction Scheme for VANET Stable Clustering. Mobile Networks and Applications. vol. 25, No. 5, 1708-1714. (2020).
14. A. Mehmood, A. Khanan, A. H. H. Mohamed, S. Mahfooz, H. Song, and S. Abdullah. ANTSC: An intelligent Naïve Bayesian probabilistic estimation practice for traffic flow to form stable clustering in VANET. IEEE Access. vol. 6, 4452-4461. (2017).
15. S. A. Karuppusamy, S. Umasangeetha, and N. Nandhagopal. Study on Intelligent Naive Bayesian Probabilistic Estimation Practice for Traffic Flow to Form Stable Clustering In VANET. International Journal Of Information and Computing Science, ISSN. vol. 6, No. 2, (2019).
16. X. Guo, Y. Chen, L. Cao, D. Zhang, and Y. Jiang. A receiver-forwarding decision scheme based on Bayesian for NDN-VANET. China Communications. vol. 17, No. 8, 106-120. (2020).
17. M.-Y. Chen, J. d. J. Rubio, and A. K. Sangaiah. Guest editorial-Pattern recognition, optimization, neural computing and applications in smart city. Computer Science and Information Systems. vol. 18, No. 4, 3-4. (2021).

18. T. Zuowen and Z. Lianfu. A review of research on privacy protection in machine learning(In Chinese). Journal of Software. vol. 31, No. 7, 2127-2156. (2020).
19. S. Alfeld, X. Zhu, and P. Barford. Data poisoning attacks against autoregressive models. in Proceedings of the AAAI Conference on Artificial Intelligence, AAAI. (2016).
20. I. M. Ahmed and M. Y. Kashmoola. Threats on Machine Learning Technique by Data Poisoning Attack: A Survey. in International Conference on Advances in Cyber Security, Springer, 586-600. (2021).
21. Z. Zhang, C. Yan, and B. A. Malin. Membership inference attacks against synthetic health data. Journal of biomedical informatics. vol. 125, No. 6, 63-81. (2022).
22. C. C. Wei Lifei, Zhang Lei, and Li Simeng. Security issues and privacy protection of machine learning. Computer Research and Development. vol. 57, No. 10, 126-148. (2020).
23. M. Nasr, R. Shokri, and A. Houmansadr. Comprehensive privacy analysis of deep learning. in 2019 ieee symposium on security and privacy, IEEE, 739-753. (2019).
24. K. Yoshida, T. Kubota, M. Shiozaki, and T. Fujino. Model-extraction attack against FPGA-DNN accelerator utilizing correlation electromagnetic analysis. in 2019 IEEE 27th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), IEEE, 318-318. (2019).
25. Y. Lindell and B. Pinkas. Privacy preserving data mining. in Annual International Cryptology Conference, Springer, 36-54. (2000).
26. A. C.-C. Yao. How to generate and exchange secrets. in 27th Annual Symposium on Foundations of Computer Science (sfcs 1986), IEEE, 162-167. (1986).
27. T. Zhang and Q. Zhu. Distributed privacy-preserving collaborative intrusion detection systems for VANETs. IEEE Transactions on Signal and Information Processing over Networks. vol. 4, No. 1, 148-161. (2018).
28. G. Raja, S. Anbalagan, G. Vijayaraghavan, S. Theerthagiri, S. V. Suryanarayan, and X.-W. Wu. SP-CIDS: Secure and Private Collaborative IDS for VANETs. IEEE Transactions on Intelligent Transportation Systems. vol. 22, No. 7, 4385-4393. (2020).
29. X. Li, H. Zhang, Y. Ren, S. Ma, B. Luo, J. Weng, et al. PAPU: Pseudonym Swap With Provable Unlinkability Based on Differential Privacy in VANETs. IEEE Internet of Things Journal. vol. 7, No. 12, 11789-11802. (2020).
30. X. Chen, T. Zhang, S. Shen, T. Zhu, and P. Xiong. An optimized differential privacy scheme with reinforcement learning in VANET. Computers & Security. vol. 110, No. 25, 1025-1056. (2021).
31. G. Raja, S. Anbalagan, G. Vijayaraghavan, P. Dhanasekaran, Y. D. Al-Otaibi, and A. K. Bashir. Energy-efficient end-to-end security for software-defined vehicular networks. IEEE Transactions on Industrial Informatics. vol. 17, No. 8, 5730-5737. (2020).
32. H. Kaur, N. Kumar, and S. Batra. ClaMPP: A cloud-based multi-party privacy preserving classification scheme for distributed applications. The Journal of Supercomputing. vol. 75, No. 6, 3046-3075. (2019).
33. T. Li, L. Lin, and S. Gong. AutoMPC: Efficient multi-party computation for secure and privacy-preserving cooperative control of connected autonomous vehicles. in SafeAI@ AAAI, CEUR Workshop Proceedings, 1-4. (2019).
34. Y. Wu, X. Wang, W. Susilo, G. Yang, Z. L. Jiang, S.-M. Yiu, et al. Generic server-aided secure multi-party computation in cloud computing. Computer Standards & Interfaces. vol. 79, No. 21, 112-130. (2022).
35. S. Sayyad. Privacy Preserving Deep Learning Using Secure Multiparty Computation. in 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), IEEE, 139-142. (2020).
36. X. Ma, F. Zhang, X. Chen, and J. Shen. Privacy preserving multi-party computation delegation for deep learning in cloud computing. Information Sciences. vol. 459, No. 2, 103-116. (2018).

37. H. Li, J. Chen, L. Wang, Q. Pei, and H. Yue. Privacy-preserving Data Aggregation for Big Data in Financial Institutions. in IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 978-983. (2020).
38. J. Zhou, S. Chen, K.-K. R. Choo, Z. Cao, and X. Dong. EPNS: Efficient Privacy Preserving Intelligent Traffic Navigation from Multiparty Delegated Computation in Cloud-Assisted VANETs. IEEE Transactions on Mobile Computing. vol. 12, No. 3, 11-25. (2021).
39. D. Ulybyshev, A. O. Alsalem, B. Bhargava, S. Savvides, G. Mani, and L. B. Othmane. Secure data communication in autonomous v2x systems. in 2018 IEEE International Congress on Internet of Things (ICIOT), IEEE, 156-163. (2018).
40. Q. Kong, R. Lu, M. Ma, and H. Bao. A privacy-preserving sensory data sharing scheme in Internet of Vehicles. Future Generation Computer Systems. vol. 92, No. 2, 644-655. (2019).
41. H. Cheng, M. Shojafar, M. Alazab, R. Tafazolli, and Y. Liu. PPVF: privacy-preserving protocol for vehicle feedback in cloud-assisted VANET. IEEE Transactions on Intelligent Transportation Systems. vol. 6, No. 12, 1-13. (2021).
42. N. Magaia, C. Borrego, P. R. Pereira, and M. Correia. ePRIVO: An enhanced privacy-preserving opportunistic routing protocol for vehicular delay-tolerant networks. IEEE Transactions on Vehicular Technology. vol. 67, No. 11, 11154-11168. (2018).
43. A. Alamer, Y. Deng, and X. Lin. A privacy-preserving and truthful tendering framework for vehicle cloud computing. in 2017 IEEE International Conference on Communications (ICC), IEEE, 1-7. (2017).
44. H. Sasaki and N. Kamiyama. Summary Cache of IoT Data Using ICN. in 2021 IFIP/IEEE International Symposium on Integrated Network Management (IM), IEEE, 707-710. (2021).
45. C. Gentry. Fully homomorphic encryption using ideal lattices. in Proceedings of the forty-first annual ACM symposium on Theory of computing, ACM, 169-178. (2009).
46. Kimlaine. Microsoft SEAL. Available: https://github.com/microsoft/SEAL
47. D. Lowd and P. Domingos. Naive Bayes models for probability estimation. in Proceedings of the 22nd international conference on Machine learning, ACM, 529-536. (2005).
48. K. M. Leung. Naive bayesian classifier. Polytechnic University Department of Computer Science/Finance and Risk Engineering. vol. 2007, 123-156. (2007).
49. A. Acar, H. Aksu, A. S. Uluagac, and M. Conti. A survey on homomorphic encryption schemes: Theory and implementation. ACM Computing Surveys (CSUR). vol. 51, No. 4, 1-35. (2018).
50. T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE transactions on information theory. vol. 31, No. 4, 469-472. (1985).
51. P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. in International conference on the theory and applications of cryptographic techniques, Springer, 223-238. (1999).
52. R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM. vol. 21, No. 2, 120-126. (1978).
53. D. Boneh, E.-J. Goh, and K. Nissim. Evaluating 2-DNF formulas on ciphertexts. in Theory of cryptography conference, Springer, 325-341. (2005).
54. J. H. Cheon, A. Kim, M. Kim, and Y. Song. Homomorphic encryption for arithmetic of approximate numbers. in International Conference on the Theory and Application of Cryptology and Information Security, Springer, 409-437. (2017).
55. Y. Yasumura, Y. Ishimaki, and H. Yamana. Secure Naïve Bayes classification protocol over encrypted data using fully homomorphic encryption. in Proceedings of the 21st International Conference on Information Integration and Web-based Applications & Services, ACM, 45-54. (2019).
56. M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz. SUMO–simulation of urban mobility: an overview. in Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation, ThinkMind, 23-28. (2011).

57. S. Mastorakis, A. Afanasyev, I. Moiseenko, and L. Zhang. A new version of the NDN simulator for NS-3. Univ. of California. 1-8. (2015).

Xian Guo is an associate professor of School of Computer and Communication, Lanzhou University of Technology. He is a visiting scholar at University of Memphis. He received MS and PhD in Lanzhou University of Technology, China, in 2008 and 2011, respectively, and BS in Nothwest Normal University. His current research interests include network and information security, cryptographic, and blockchain. E-mail: iamxg@ 163.com.

Baobao Wang is currently a master student at Computer and Communication School of Lanzhou University of Technology. He received his Bachelor degree from North China Institute of Science and Technology, China, in 2019, and started his master studying in 2019. His research interests are machine learning, Information-Centric Networking etc., Secure Multiparty Computation.

Yongbo Jiang is a lecturer of School of Computer and Communication, Lanzhou University of Technology. He received MS and PhD in Xidian University, China, in 2008 and 2013, respectively. His current research interests include network and information security, and Information-Centric Networking etc.

Di Zhang is an associate professor of School of Computer and Communication, Lanzhou University of Technology. He received MS and PhD in Communication University of China, China, in 2013 and 2016, respectively. His current research interests include network and information security, and blockchain etc.

Laicheng Cao is a professor of School of Computer and Communication, Lanzhou University of Technology. He received MS in Lanzhou University, China, in 2004. His current research interests include network and information security, and cryptography etc.

Received: February 10, 2022; Accepted: July 02, 2022.