

Received July 24, 2020, accepted August 8, 2020, date of publication September 2, 2020, date of current version October 7, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3021253

Blockchain-Based Federated Learning for Intelligent Control in Heavy Haul Railway

GAOFENG HUA¹, LI ZHU¹, JINSONG WU^{1,2,3}, (Senior Member, IEEE), CHUNZI SHEN¹, LINYAN ZHOU¹, (Student Member, IEEE), AND QINGQING LIN⁴

¹State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China

²School of Artificial Intelligence, Guilin University of Electronic Technology, Guilin 510000, China

³Department of Electrical Engineering, Universidad de Chile, Santiago 8370451, Chile

⁴Huawei Beijing Research Institute, Beijing 100095, China

Corresponding author: Jinsong Wu (wujs@ieee.org)

This work was supported in part by the National Natural Science Foundation of China under Grant 61973026, in part by the Beijing Science and Technology Commission Funding under Grant Z191100010818001, in part by the Beijing Education Commission Funding under Grant I20H100010 and Grant I19H100010, in part by the Beijing Natural Science Foundation under Grant L181004, in part by the Fundamental Research Funds for the Central Universities under Grant 2018JBZ002, in part by the Hunan Provincial Nature Science Foundation under Grant 2018JJ2535, in part by the Chile Comisión Nacional de Investigación Científica y Tecnológica (CONICYT) Fondo Nacional de Desarrollo Científico y Tecnológico (FONDECYT) Regular under Grant 1181809, and in part by the Chile CONICYT Fondo de Fomento al Desarrollo Científico y Tecnológico (FONDEF) under Grant ID16I10466.

ABSTRACT Due to the long train marshaling and complex line conditions, the operating modes in heavy haul rail systems frequently change when trains travel. Improper traction or braking operation made by drivers will increase the longitudinal impact force to trains and causes the train decoupling, severely affecting the safe operations of trains. It is quite desirable to replace the manual control with intelligent control in heavy haul rail systems. Traditional machine learning-based intelligent control methods suffer from insufficient data. Due to lacking effective incentives and trust, data from different rail lines or operators cannot be shared directly. In this paper, we propose an approach on blockchain-based federated learning to implement asynchronous collaborative machine learning between distributed agents that own data. This method performs distributed machine learning without a trusted central server. The blockchain smart contract is used to realize the management of the entire federated learning. Using the historical driving data collected from real heavy haul rail systems, the learning agent in the federated learning method adopts a support vector machine (SVM) based intelligent control model. To deal with the imbalanced traction and braking data, we optimize the classic SVM model via assigning different penalty factors to the majority and minority classes. The data set are mapped to a high dimension using kernel functions to make it linearly separable. We construct a mixing kernel function composed of polynomial and radial basis function (RBF) kernel functions, which uses a dynamic weight factor changing with train speeds to improve the model accuracy. The simulation results demonstrate the efficiency and accuracy of our proposed intelligent control method.

INDEX TERMS Federated learning, blockchain, support vector machine, radial basis function, heavy haul railway.

I. INTRODUCTION

The heavy haul railway has the advantages of large transportation capacity, high efficiency, low energy consumption, and low transportation cost, which has attracted attention from all over the world and has been worldwide acknowledged as the main development direction for railway bulk cargo transportation.

The associate editor coordinating the review of this manuscript and approving it for publication was Jun Wu¹.

Due to the long train marshaling, and complex line conditions, the operating mode in heavy haul rail systems frequently changes when trains travel. An improper traction or braking operation made by drivers will increase the longitudinal impact force to trains and causes the train decoupling, severely affecting the safe operation of trains. It is quite desirable to replace the manual control with intelligent control in heavy haul rail systems.

To realize the safe and efficient control of trains, scholars from various countries have studied related theories and

applications in different fields. Traditional Train control algorithms mainly include proportional integral derivative (PID) classical control theory, fuzzy control, and machine learning. The PID control algorithm controls the train operation by calculating the various operating conditions. Zhuan used an open-loop controller to determine the power distribution between the front and rear locomotives and track the target curve in conjunction with a closed-loop controller [1]. Grube and Bayoumi implemented curve tracking to minimize coupler forces caused by disturbances such as slope [2]. Neural networks learn the rules of data hiding via designing neural network models to achieve the purpose of prediction [3]. Bai introduced a fuzzy neural network to implement intelligent control for freight train docking based on historical data of train docking stations [4]. Dewang Chen combined a linear model, a generalized regression neural network, and a fuzzy inference system to estimate the parking error of urban rail transit trains, and then dynamically optimized and adjusted the parking error [5]. Reinforcement learning, as a variant of Markov's decision-making process [6], has also demonstrated satisfactory train control characteristics. Li Zhu used deep reinforcement learning to optimize the communication performance jointly and train control strategy based on the channel characteristics of the communications-based train control (CBTC) based communication system and real-time train position information [7]. Dewang Chen modeled train control into a multi-stage decision-making process based on the transponder positioning information, and set the reciprocal of the parking error as the reward value, and introduced reinforcement learning to solve the maximum reward function [8]. Besides, the work [8] used the Markov decision process (MDP) to model the driving behavior of urban rail transit drivers, constructed a return function through multiple indicators, and applied the Q-Learning algorithm to solve it to achieve online train control [6].

One crucial problem in the above-mentioned existing works is their assumptions about the dynamic train model. The traction and braking systems of heavy-haul trains are typical nonlinear time-varying systems that are difficult to describe with accurate mathematical models such as PID control. Machine learning algorithms are widely used today, such as in agriculture [9], bioinformatics [10], [11], and wireless communications [12]. The machine learning algorithm model represented by a neural network has strong self-adaptability and nonlinear processing ability but has the disadvantages of slower convergence speed, optimal local solution, and easy over-fitting. A straightforward approach to conduct machine learning is first collecting and storing the data in one central server, and then processing them all together [13], [14].

Furthermore, traditional methods on machine learning-based intelligent control suffer from insufficient data. Considering the data privacy and security, data from different railway lines or operators cannot be directly shared. Data barriers between operators have severely hindered the development of intelligent rail transits. There seem to be uncoordinated contradictions between the exchange of data and the security

of data. It is to be solved how to connect the islands of data fragmentations without revealing the privacy as well as realize the sharing of data and the co-construction of models.

In 2016, Google first proposed the concept of federated learning [15], [16]. Federated learning is to build machine learning models based on data sets that are distributed across multiple devices while preventing data leakage. Recent relevant improvements have been focusing on overcoming the statistical challenges [17], [18] and improving security [19], [20] in federated learning. There are also research efforts to make federated learning more personalizable [21], [17]. Federated learning can effectively solve the problem of insufficient data and protect the data privacy and security.

The combination of federated learning (FL) and blockchain is a hot topic of recent research. Y. Lu proposed a new architecture based on federated learning to relieve transmission load and address privacy concerns of providers in Internet of vehicles [22]. K. Toyoda proposed to introduce repeated competition for FL so that any rational worker follows the protocol and maximizes its profits [23]. Y. Lu first designed a blockchain empowered secure data sharing architecture for distributed multiple parties to protect security and privacy of sharing data in wireless networks [24]. H. Kim proposed a blockchained federated learning architecture where local learning model updates are exchanged and verified, which enables on-device machine learning without any centralized training data or coordination via utilizing a consensus mechanism in blockchain [25]. Sana Awan proposed a blockchain-based privacy-preserving federated learning framework, which leverages the immutability and decentralized trust properties of blockchain to provide the provenance of model updates [26].

For the first time, we apply joint federated learning and blockchain to the heavy haul rail systems to protect the data privacy and security of operators. In this paper, we propose a federated learning framework based on blockchain, which enables different operators to train intelligent driving models without sharing data. Operators do not necessarily share their private data, but only necessarily train their intelligent driving models locally and share the training weights through the blockchain. They would obtain the final intelligent train driving model through our proposed federated learning method. Blockchain-based federated learning can protect the data privacy of operators and train intelligent driving models more accurately than single operator training. We introduce a support vector machine (SVM) based intelligent control model for the learning agent in the proposed distributed federated learning method. The unbalanced traction and braking data are handled via assigning different punishment factors to the main category and a few categories. The kernel function is introduced to map the data set to a high dimension to make them linearly separable. The performance difference between the polynomial kernel function and the radial basis function (RBF) kernel function in different scenarios is compared. The dynamic update factor is generated via combining the train running speed, and then the

algorithm is optimized to improve the data recognition for the mode.

The rest parts of this paper are summarized as follows. In section II, we describe the federated learning framework based on blockchain. In section III, we model of heavy haul train traction and propose electric braking based on SVM. Section IV gives the SVM algorithm optimization results, and section V gives the conclusion.

II. BLOCKCHAIN-BASED FEDERATED MACHINE LEARNING FRAMEWORK

In this section, we propose a federated learning framework based on blockchain, which is decentralized and privacy-preserving and enables each operator to train our intelligent driving model without leaking their private data.

A. BLOCKCHAIN

Blockchain, proposed by Nakamoto in 2008 for Bitcoin [27], is the cornerstone of the modern digital cryptocurrency system. In the last few years, academia and industry have conducted extensive research efforts on blockchain and found that this advanced technology can be applied to applications in various fields (such as finance, healthcare, and asset registration).

Blockchain is a chronologically ordered list of blocks, where each block, identified by its unique cryptographic hash, refers to the block came before it, resulting in a chain of blocks. Once a block is created and appended to the blockchain, the transaction information in that block cannot be changed or reverted, which ensures the integrity of the system.

We use blockchain to store, transfer and share machine learning models. A very critical technology in the blockchain is smart contracts. The term smart contract was first proposed by Nick Szabo in 1995. In several articles he published, he defined smart contracts as: "A smart contract is a set of promises defined in digital form. Including agreements on which contract participants can execute these commitments." [28]. But at the time there were no digital systems and technologies that could support programmable contracts. Work on smart contracts could only remain at the theoretical stage, after the advent of blockchain Smart contracts can be applied in practice. The blockchain is suitable for programmable contracts, and its distributed, non-tamperable and traceable characteristics are very consistent with smart contracts, so smart contracts have quickly become one of the characteristics of blockchain technology [29]. We use blockchain smart contracts to implement a series of operations such as system initialization, information interaction, training timing, and data storage.

The blockchain realizes the automatic execution of federated learning through smart contracts. The entire process can be traced back, not tampered with, and decentralized. Smart contracts based on blockchain technology can take advantage of the cost-efficiency of smart contracts and avoid the interference of malicious behavior with the normal execution of

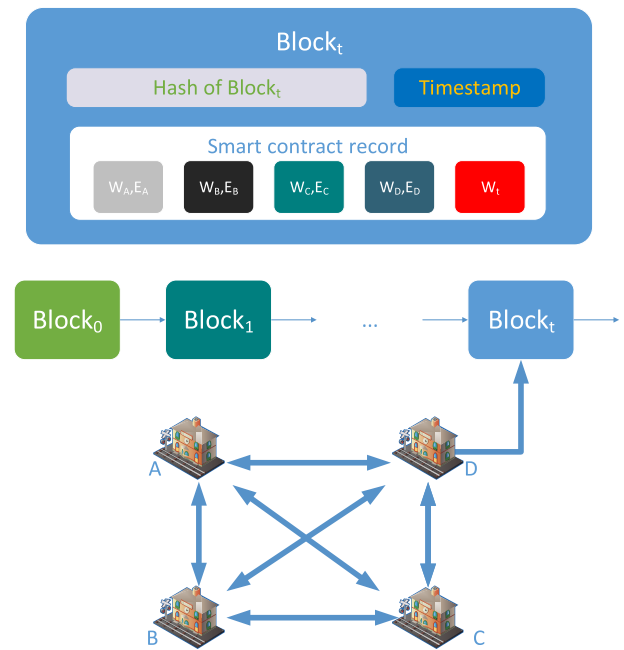


FIGURE 1. Federated learning framework based on blockchain.

contracts. The smart contract is written into the blockchain in a digital form, and the characteristics of the blockchain technology guarantee the storage, reading, and execution of parameters. The entire learning process is transparent, traceable, and unchangeable.

B. THE FRAMEWORK OF FEDERATED LEARNING BASED ON BLOCKCHAIN

We describe the federated learning framework based on blockchain in Fig. 1. We assume that four operators have their private data. They have the same data type, but their data sizes are limited. Their common goal is to train a general intelligent control model for heavy haul trains. We call these operators learning participants. The entire federated learning process includes the following processes:

- The smart contract will act as the executor of the blockchain to automatically realize the iterative federated learning process. First, all participants participate in the formulation of a smart contract. Then the smart contract is spread through the P2P network and stored in the blockchain. Finally, the smart contract in the blockchain will automatically execute the learning process.
- Learning participants A, B, C, and D calculate their model parameter w based on the current model, encapsulate and broadcast all nodes together, and the corresponding model's error rate.
- All learning participants compete to obtain permission to add new blocks to the chain via solving mathematical puzzles. After a learning participant obtains the permission to produce blocks, it collects all the model parameter w and update the model parameter w in blockchain. Finally, the new block 'Block_t' are generated into the blockchain. 'Block_t' contains the

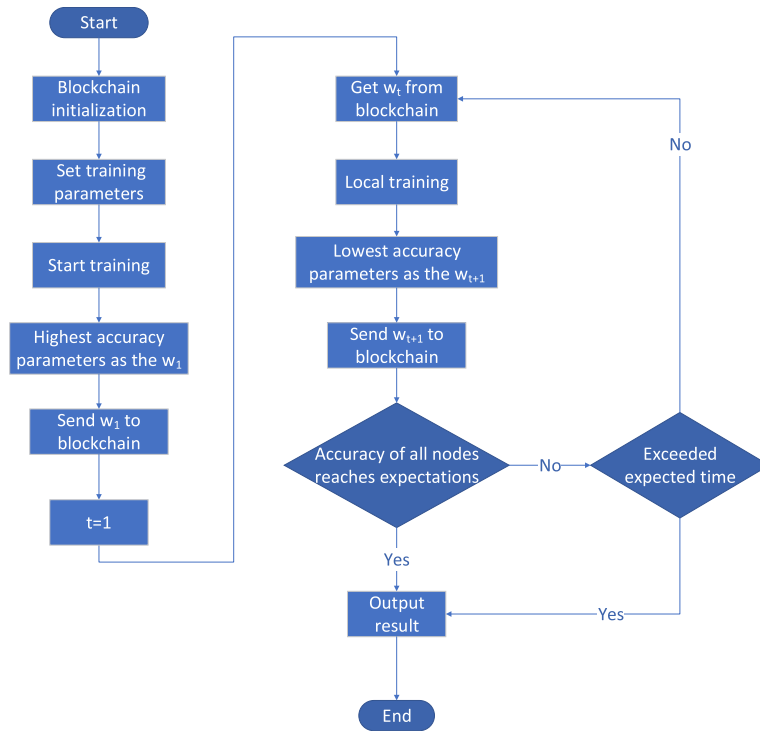


FIGURE 2. Flowchart for federated learning based on blockchain.

hash value, time, and transaction information of the block.

- Optimizing parameters w is the most critical part of federated learning. It is also a parameter shared in model training. Federated learning implements model training by continuously iterating to optimize this parameter. The optimization parameters represent different meanings in different learning processes. For details, please refer to Chapter IV, Section 4. Since participants only share the learning parameter w without sharing data during the learning process, the data privacy of the participants can be protected.

C. FEDERATED LEARNING BASED ON BLOCKCHAIN

The method of federated learning process is shown in Fig. 2, as follows:

- 1) Initialization stage: Set training parameters ($t = 1$), including the parameters to be tuned, initial point, step size, search range, predetermined accuracy rate, and predetermined training period.
- 2) At the initial stage ($t = 1$), the four participants A, B, C, and D use the preset initial point, step size, and search range to search for the optimal parameter w through the grid optimization method.
- 3) Taking A as an example, assume that the model trained by the parameters found by A has the highest accuracy. In order to prevent the error break from being propagated during the initialization phase, we select the parameter w_A found by A as the central training

parameter w_1 for the next cycle of training, and the participant who obtains the permission to produce block is responsible for collecting the current round of training parameters of all nodes and uploading it to the blockchain to form $Block_1$. Based on this, other nodes perform the next training according to the preset step size and search range.

- 4) In the next stage ($t = 2$), in order to improve the accuracy of all nodes, A, B, C, D will obtain the central training parameter w_1 of the previous stage from the blockchain, and center on w_1 with a preset step size and search range to search the optimal parameters through grid optimization method. The node with the lowest training accuracy rate in this round provides the central training parameter w_2 . The participant who obtains the permission to produce block collects the training parameters of all nodes in this round, and uploads it to the blockchain to form $block_2$. For the next iteration process.
- 5) Until the accuracy rates of the four nodes A, B, C, and D all exceed the preset value, or the training time exceeds the preset period, the training ends, and the training result is returned.

In the federation learning process, only the optimized parameters are shared without sharing the original data, which protects the user's privacy from the source. The combination of federated learning and blockchain guarantees security and non-tampering in the learning process, thus protecting user data privacy and security.

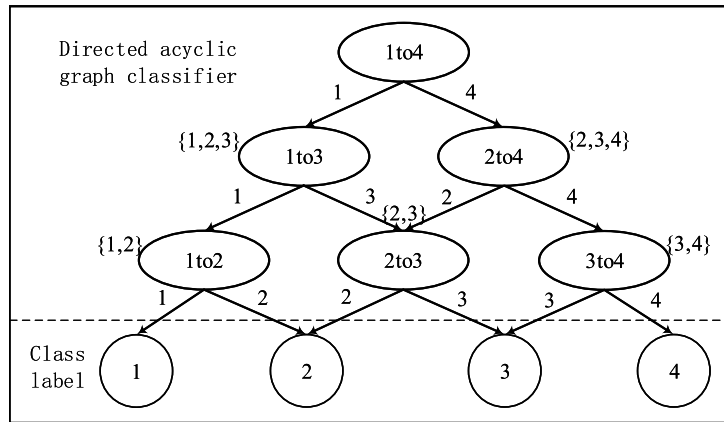


FIGURE 3. Structure diagram of directed acyclic graph.

III. MODELING OF HEAVY HAUL TRAIN TRACTION AND ELECTRIC BRAKING BASED ON SVM

This paper takes the SS4G heavy haul train as the research object, a stepped traction locomotive including traction gears, brake gears, and a coasting gear. Hence, the problem in the intelligent control of traction and electric braking force for heavy haul trains is transformed into a classification problem of machine learning. The SVM algorithm is used to establish a classification model to implement the intelligent control of heavy haul train traction and electric braking.

A. CONVERSION FROM BINARY CLASSIFICATION MODEL TO MULTI-CLASSIFICATION MODEL

The SVM algorithm is a typical binary classification algorithm to construct a hyperplane to identify the data. For heavy-haul trains, there are 17 gears for traction, electric braking, and inertia, which requires a multi-classification model to achieve the intelligent control of them.

The first layer of the model determines whether to coast. The second layer of the model determines the output traction or electric braking. After the predicted data are judged by the first-layer decision-maker to not output inertia, the output traction or electric braking will be determined through this layer model.

The above two layers have obvious features in determining the driving strategy, and each layer only needs a classifier to achieve better results. However, for the discrimination of specific gears for traction or electric braking, the features have fewer differences, and drivers' driving habits are different. To simulate the driving strategies of excellent drivers as many as possible, this paper uses the following directed acyclic graph method for traction or the selection of specific gears for electric braking for multi-class modeling.

The idea of directed acyclic graphs is to design an SVM model for any two types of labels. Therefore, for a dataset with k categories, $k(k - 1)/2$ classifiers need to be constructed. We assume a total of 4 categories {1,2,3,4} in the data set, via constructing a directed acyclic graph as shown in Fig. 3, the $k(k - 1)/2$ classifiers are combined to form the final multi classifier. It can be seen from the structure

of the model that the model has $(k - 1)$ layers, that is, for the data to be predicted, only $(k - 1)$ models are needed to obtain the final result. The prediction response time of this method is significantly better than that of all classifiers. The method has better time performance. The disadvantage of directed acyclic graphs is that a large number of models need to be trained, and the training time is longer. This disadvantage will not have a negative effect on the intelligent driving of heavy-load railways. Therefore, the design method of the directed acyclic graph has a good application scenario.

The third layer will be constructed according to the structure of the directed acyclic graph described above. The data labeled with traction and electric braking will be applied to the training of the corresponding model. As shown in Fig. 4, it is the overall structure of the intelligent controller. The disadvantage of the directed acyclic graph is that the errors at the upper layer will accumulate downward. In order to reduce the degree of error accumulation, each binary classifier in the directed acyclic graph is designed to realize the classification of the two categories with the greatest difference. For example, the first level of the directed acyclic graph of traction gears is the 1st gear and the 10th gear decision. It can be seen from Fig. 4 that the model ultimately needs to construct 62 binary classifiers.

B. SELECTION OF SVM KERNEL FUNCTIONS

For the case where the data set are nonlinear, SVM's solution is to introduce a kernel function. By mapping the data to a linearly separable high-dimensional space, SVM is used to perform linear classification in the high-dimensional space. The design of the kernel function is an essential factor that affects the performance of the algorithm. A detailed description of the application for the kernel function will be given in this section.

Mapping data from a linearly indivisible low-dimensional space to a linearly separable high-dimensional space requires a nonlinear transformation of the feature x of the original data. We assume that the transformed feature is $z = \varphi(x)$, the decision function of SVM in the new feature space is

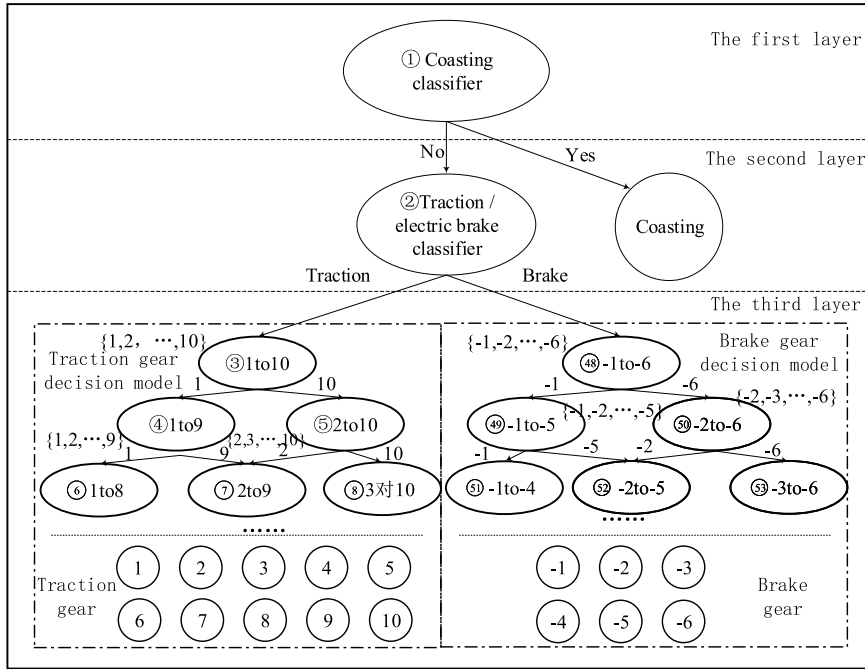


FIGURE 4. Structure diagram of heavy haul train traction and electric brake classifiers.

formed as:

$$g(x) = \text{sgn}\left\{\sum_{i=1}^n y_i \alpha_i^* [\varphi(x_i) * \varphi(x)] + b^*\right\}. \quad (1)$$

The kernel function is introduced, which is given by

$$K(x_i, x_j) = [\varphi(x_i) * \varphi(x_j)]. \quad (2)$$

The nonlinear data set are implicitly mapped to a high-dimensional space via replacing the inner product's expression with a suitable kernel function. The SVM model's decision function that finally introduces the kernel function is obtained via substituting (2) into (1).

The construction and selection of the kernel function will directly influence the effect of the training for the SVM model in high-dimensional space. The kernel functions are given as follows:

- 1) The polynomial kernel function is equivalent to:

$$K(x_i, x) = [(x_i) * x + 1]^d. \quad (3)$$

- 2) Gauss radial basis kernel function (RBF kernel function) is formed as:

$$K(x_i, x) = \exp\left\{-\frac{\|x_i - x\|^2}{\sigma^2}\right\}. \quad (4)$$

From the experience of the application using kernel functions in SVM, it can be known that different kernel functions have important impacts on the effects of SVM models, mainly in the selections of penalty factors and the parameters of kernel functions themselves. Two kernel functions are optimized via parameter tuning. The final results are used to construct a mixed kernel function to optimize the model.

C. IMPROVEMENT OF THE SVM MODEL BASED ON THE MIXED KERNEL FUNCTION

As observed from the previous section, the two kernel functions show the performance differences in two different application scenarios. The greatest difference between the two scenarios is the change of the train speed. In most of the cases, the acceleration is small in the coasting mode and is large in the traction or electric braking mode.

According to the analysis of the above results, it can be known that the RBF kernel function has a good classification effect on traction and electric braking, and the SVM model introduced with a polynomial kernel function has higher accuracy in predicting whether to run idle. Thus, the following optimization measures are proposed. The two are combined via increasing the adaptive factor, which is given by

$$K(x_i, x) = \beta * \exp\left\{-\frac{\|x_i - x\|^2}{\sigma^2}\right\} + (1 - \beta) * [(x_i) * x + 1]^d, \quad (5)$$

where $\beta \in [0,1]$ denotes the adaptive weighting factors for two kernel functions. To make the full use of the speed change information of the train, the speed of the train is integrated into adaptive weighting factors, which is equivalent to

$$\beta = 1 - \exp(-|k|), \quad (6)$$

where k denotes the change in the speed of train adjacent control cycles, namely, the acceleration, which is formed as

$$k = \frac{v_i - v_{i-1}}{t_i - t_{i-1}}. \quad (7)$$

When $|k|$ is large, that is, the train acceleration is large, β is relatively high, and the RBF kernel function occupies a larger proportion in the spatial map. Conversely, the smaller $|k|$ is, the smaller the train acceleration will be, and the polynomial kernel function will have a larger proportion in spatial mapping. Thus, through the adaptive adjustment of β , the mixed kernel function achieves the original design purpose. The hybrid kernel function will be introduced in the third layer of the model.

D. PARAMETER TUNING FOR SVM MODEL

The parameters of the support vector machine model consist of two parts. The first part is the inherent parameters of the model, namely, the penalty factors C_+ of the SVM; the second part is the parameters carried in the kernel function. The second part varies with different kernel functions. The key parameters of the polynomial kernel function and RBF kernel function are described as follows:

The key parameter of the polynomial kernel function is the highest degree term d . The larger d is, the higher the dimensionality of the mapping will be. However, as d grows, the operation complexity grows exponentially, which will easily cause the training server to terminate the task early. The key parameter of the RBF kernel function is the kernel width σ . The parameter determines the complexity of the sample data; that is, the complexity of the data set after the samples are mapped.

In the process of federated learning, parameter tuning is a very important step. The three-layer classifier will be federated independently for three times to obtain tuning parameters.

- 1) For the first-level lazy row classifier, this paper will use a polynomial kernel function to optimize the parameters. In this step, the parameters to be tuned by the SVM are the SVM penalty factor C_+ and the maximum degree d of the polynomial kernel function.
- 2) For the second layer of traction and electric brake classifiers, this paper will use the RBF kernel function to optimize parameters. In this step, the parameters that SVM needs to tune are SVM penalty factor C_+ and RBF kernel function kernel width σ .
- 3) For the third-level classifier, this article will use the mixed kernel function mentioned in this chapter for parameter tuning. The parameters to be adjusted for the mixed kernel function include the kernel width σ of the RBF kernel function and the highest order d of the polynomial kernel function.

We will use the second chapter of the federated learning method to separately train the three classifiers to obtain the most suitable SVM model. This article sets the predetermined training period to 50, and the predetermined accuracy rate is 95% through a preliminary exploration of the parameters. When the accuracy of the four nodes A, B, C, and D reaches 95% or exceeds the predetermined training period, the training ends. For the first-level classifier, the initial point of C_+ is

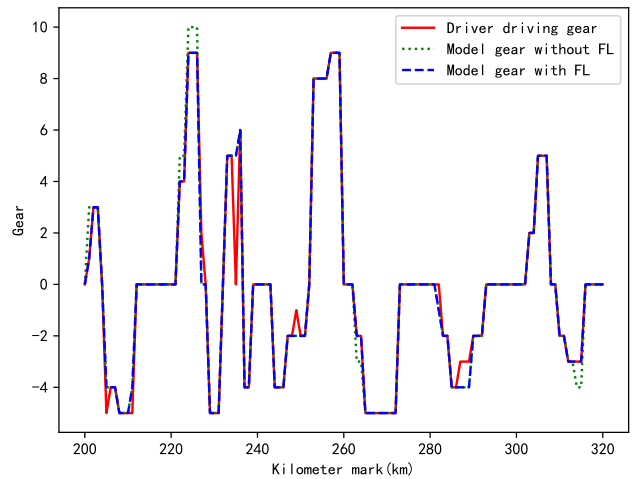


FIGURE 5. Comparison of SVM model control results and driver driving results.

500, the search step is 50, the initial order of the polynomial kernel function is 1, and the step is 1. The initial point of the second-level classifier C_+ is 500, and the search step is 50, the RBF kernel function width is initially 0, and the search step is 0.5; the third layer classifier second layer classifier C_+ initial point is 500, the search step is 50, the polynomial kernel function initial order is 1, and the step is 1. The RBF kernel function width is initially 0, and the search step is 0.5.

IV. PERFORMANCE

This section introduces the performance evaluation of the proposed approach through the simulation results. The first subsection introduces the comparison of the training results of A participant data and A, B, C, D federation after learning. The second subsection introduces the SVM simulation results of three different classifiers.

A. FEDERATED LEARNING RESULTS

To verify the accuracy of the intelligent control model and federated learning, this paper selects the train running data from Xiaojue Station of Shuohuang Railway to West Station of Dingzhou to verify the model accuracy. The total length of the section is 120 kilometers, and the maximum downhill gradient in the section is less than -4% , and the terrain is gentle. During the train operation, the driver mainly relies on traction and electric braking force to adjust the train speed, which is conducive to model verification. The model control results and the actual driving results of the driver are shown in Fig. 5. The red line represents the driver's driving gear. The green line represents the model control gear trained by A's data, and the blue line represents the model control gear with federated learning. The overall prediction accuracy without federated learning is 84.30%. The overall prediction accuracy with federated learning is 94.21%.

B. RESULTS OF SVM MODEL

- 1) Coasting classifier:

The Optimization results of the coasting classifier is shown in Fig. 6. The first layer uses a polynomial kernel

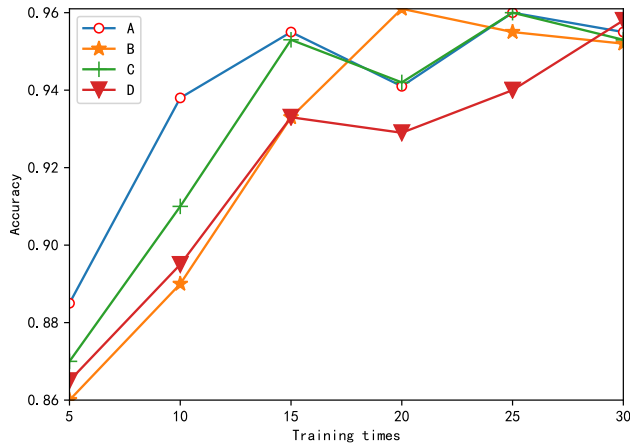


FIGURE 6. Optimization results of the coasting classifier.

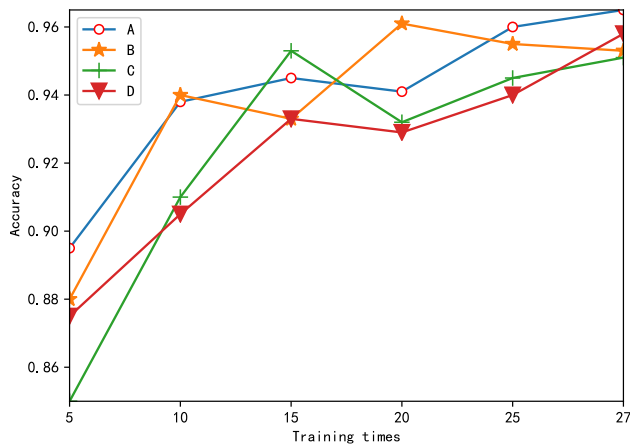


FIGURE 7. Optimization results of traction or braking classifier.

function. As can be seen from the figure, as the training period increases, the training accuracy of the four nodes A, B, C, and D is steadily increasing. When the training period reaches 30, the accuracy rates of A, B, C, and D all exceed 95%. At this point, the training is over.

2) Traction/electric brake classifier:

The Optimization results of traction or braking classifier is shown in Fig. 7. The second layer uses the RBF kernel function. As can be seen from the figure, as the training period increases, the training accuracy of the four nodes A, B, C, and D is steadily increasing. When the training period reaches 27, the accuracy rates of A, B, C, and D all exceed 95%. At this point, the training is over.

C. RESULTS OF THE SVM MODEL BASED ON THE MIXED KERNEL FUNCTION

The optimized results of decision in the highest and lowest gear for traction or braking is shown in Fig. 8. The third layer uses a mixed kernel function. As can be seen from the figure, as the training period increases, the training accuracy of the four nodes A, B, C, and D is steadily increasing. When the training period reaches 32, the accuracy rates of

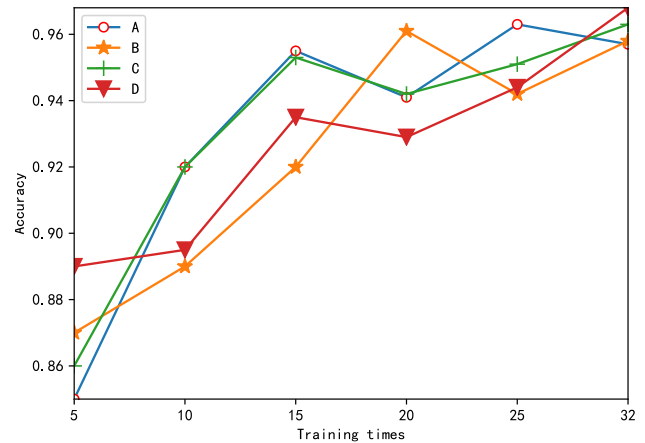


FIGURE 8. Optimized results of decision in the highest and lowest gear for traction or braking.

all participants exceed 95%. At that time, the training is over.

V. CONCLUSION

In this paper, we have proposed a blockchain-based federated learning framework to protect user data privacy and security. This method performs distributed machine learning without a trusted central server. The blockchain smart contract is used to realize the management of the entire federated learning. Based on this, the SVM classification model has been introduced to realize the intelligent control of traction/electric braking of heavy haul trains. We have introduced the directed acyclic graph, which helps the migration of two classification models to multiple classification models. Then, by comparing the SVM model's prediction effect with the polynomial kernel function and the RBF kernel function on the data set, we have analyzed the features of the two in different operating scenarios. In coasting scenarios, the polynomial kernel function performance is better. In traction/electric brake scenarios, the RBF kernel function performance is better. Then, a dynamic update factor has been constructed in combination with the train's speed, and the two have been connected to form a hybrid kernel function to optimize the algorithm. Distributed machine learning has been performed through the federated learning framework of heavy haul trains. An intelligent control model for heavy haul trains has been obtained through a fusion algorithm. Finally, the optimal intelligent control model has been used to predict the interval operation data. The comparisons between federated learning results and other machine learning comparison results have demonstrated that this federated learning method can train better model by more data. The comparison with the driver driving results has demonstrated the effectiveness of the intelligent control model.

REFERENCES

- [1] X. Zhuan, "Optimal handling and fault tolerant speed regulation of heavy haul trains," Ph.D. dissertation, Fac. Eng., Built Environ. Inf. Technol., Univ. Pretoria, Pretoria, South Africa, 2007.

- [2] P. Gruber and M. M. Bayoumi, "Suboptimal control strategies for multi locomotive powered trains," *IEEE Trans. Autom. Control*, vol. AC-27, no. 3, pp. 536–546, Jun. 1982.
- [3] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989.
- [4] Y. Bai, B. Mao, T. Ho, Y. Feng, and S. Chen, "Station stopping of freight trains with pneumatic braking," *Math. Problems Eng.*, vol. 2014, pp. 1–7, Jul. 2014.
- [5] D. Chen and C. Gao, "Soft computing methods applied to train station parking in urban rail transit," *Appl. Soft Comput.*, vol. 12, no. 2, pp. 759–767, Feb. 2012.
- [6] J. Yin, D. Chen, and L. Li, "Intelligent train operation algorithms for subway by expert system and reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 6, pp. 2561–2571, Dec. 2014.
- [7] L. Zhu, Y. He, F. R. Yu, B. Ning, T. Tang, and N. Zhao, "Communication-based train control system performance optimization using deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 10705–10717, Dec. 2017.
- [8] D. Chen, R. Chen, Y. Li, and T. Tang, "Online learning algorithms for train automatic stop control using precise location data of balises," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1526–1535, Sep. 2013.
- [9] R. J. McQueen, S. R. Garner, C. G. Nevill-Manning, and I. H. Witten, "Applying machine learning to agricultural data," *Comput. Electron. Agricult.*, vol. 12, no. 4, pp. 275–293, Jun. 1995.
- [10] X. Chen, J. Ji, T. Ji, and P. Li, "Cost-sensitive deep active learning for epileptic seizure detection," in *Proc. ACM Int. Conf. Bioinf., Comput. Biol., Health Informat.*, Aug. 2018, pp. 226–235.
- [11] X. Chen, J. Ji, K. Loparo, and P. Li, "Real-time personalized cardiac arrhythmia detection and diagnosis: A cloud computing architecture," in *Proc. IEEE EMBS Int. Conf. Biomed. Health Informat. (BHI)*, Orlando, FL, USA, Feb. 2017, pp. 201–204.
- [12] C. Luo, J. Ji, Q. Wang, X. Chen, and P. Li, "Channel state information prediction for 5G wireless communications: A deep learning approach," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 1, pp. 227–236, Jan. 2020.
- [13] V. S. Verykios, E. Bertino, I. N. Fovino, L. P. Provenza, Y. Saygin, and Y. Theodoridis, "State-of-the-art in privacy preserving data mining," *ACM Sigmod Rec.*, vol. 33, no. 1, pp. 50–57, 2004.
- [14] X. Wu, X. Zhu, G.-Q. Wu, and W. Ding, "Data mining with big data," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 1, pp. 97–107, Jan. 2014.
- [15] J. Konecny, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *CoRR*, vol. abs/1610.02527, pp. 1–38, Oct. 2016. [Online]. Available: <http://arxiv.org/abs/1610.02527>
- [16] J. Konecny, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *CoRR*, vol. abs/1610.05492, pp. 1–10, Oct. 2016. [Online]. Available: <http://arxiv.org/abs/1610.05492>
- [17] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, Inc., 2017, pp. 4424–4434. [Online]. Available: <http://papers.nips.cc/paper/7029-federated-multi-task-learning.pdf>
- [18] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data," 2018, *arXiv:1806.00582*. [Online]. Available: <https://arxiv.org/abs/1806.00582>
- [19] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, New York, NY, USA, Oct. 2017, pp. 1175–1191, doi: [10.1145/3133956.3133982](https://doi.org/10.1145/3133956.3133982).
- [20] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," *CoRR*, vol. abs/1712.07557, pp. 1–7, Mar. 2017. [Online]. Available: <http://arxiv.org/abs/1712.07557>
- [21] F. Chen, Z. Dong, Z. Li, and X. He, "Federated meta-learning for recommendation," *CoRR*, vol. abs/1802.07876, pp. 1–14, Dec. 2018. [Online]. Available: <http://arxiv.org/abs/1802.07876>
- [22] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Blockchain empowered asynchronous federated learning for secure data sharing in Internet of vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4298–4311, Apr. 2020, doi: [10.1109/TVT.2020.2973651](https://doi.org/10.1109/TVT.2020.2973651).
- [23] K. Toyoda and A. N. Zhang, "Mechanism design for an incentive-aware blockchain-enabled federated learning platform," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Los Angeles, CA, USA, Dec. 2019, pp. 395–403, doi: [10.1109/BigData47090.2019.9006344](https://doi.org/10.1109/BigData47090.2019.9006344).
- [24] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020, doi: [10.1109/TII.2019.2942190](https://doi.org/10.1109/TII.2019.2942190).
- [25] H. Kim, J. Park, M. Bennis, and S. Kim, "Blockchained on-device federated learning," *IEEE Commun. Lett.*, vol. 24, no. 6, pp. 1279–1283, Jun. 2020, doi: [10.1109/LCOMM.2019.2921755](https://doi.org/10.1109/LCOMM.2019.2921755).
- [26] S. Awan, F. Li, B. Luo, and M. Liu, "Poster: A reliable and accountable privacy-preserving federated learning framework using the blockchain," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Nov. 2019, pp. 2561–2563, doi: [10.1145/3319535.3363256](https://doi.org/10.1145/3319535.3363256).
- [27] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Tech. Rep., 2008.
- [28] T. Bocek, *Digital Marketplaces Unleashed*. New York, NY, USA: Springer-Verlag GmbH, 2017, pp. 169–184.
- [29] D. Tapscott and A. Tapscott, *Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World*. London, U.K.: The Economist, May 2016, p. 72, 83, 101, and 127.



GAOFENG HUA received the B.S. degree in electronic and information engineering from Beijing Jiaotong University, Beijing, China, in 2019, where he is currently pursuing the master's degree with the State Key Laboratory of Rail Traffic Control and Safety. His research interests include blockchain, federated learning, and communication-based train control (CBTC) systems.



LI ZHU received the Ph.D. degree in traffic control and information engineering from Beijing Jiaotong University, Beijing, China, in 2012. He is currently a Faculty Member with Beijing Jiaotong University. He is also a Visiting Scholar with Carleton University, Ottawa, ON, Canada, and the University of British Columbia, Vancouver, BC, Canada. His research interests include train-ground communication technology in communication-based train ground communication (CBTC) systems and cross layer design in train-ground communication systems.



JINSONG WU (Senior Member, IEEE) received the Ph.D. degree from the Department of Electrical and Computer Engineering, Queen's University, Canada. He is elected as the Vice-Chair, Technical Activities, IEEE Environmental Engineering Initiative, a pan-IEEE effort under IEEE Technical Activities Board (TAB). He was the Founder and the Founding Chair of the IEEE Technical Committee on Green Communications and Computing (TCGCC). He is also the Co-Founder and the Founding Vice-Chair of the IEEE Technical Committee on Big Data (TCBD). He coauthored paper won 2018 IEEE TCGCC Best Magazine Paper Award. He received the 2017 and 2019 IEEE System Journal Best Paper Awards and the 2017 IEEE Green Communications and Computing Technical Committee Excellent Services Award for Excellent Technical Leadership and Services in the Green Communications and Computing Community. He was the Leading Editor and the coauthor of the comprehensive book, entitled *Green Communications: Theoretical Fundamentals, Algorithms, and Applications* (CRC Press, September 2012).



CHUNZI SHEN received the B.S. degree in electronic and information engineering from Beijing Jiaotong University, Beijing, China, in 2019, where she is currently pursuing the master's degree with the State Key Laboratory of Rail Traffic Control and Safety.



QINGQING LIN received the master's degree in electronic and information engineering from Beijing Jiaotong University, Beijing, China, in 2019. He is currently working with Huawei Technologies Company Ltd. His research interest includes rail transit information engineering and control.

...



LINYAN ZHOU (Student Member, IEEE) is currently pursuing the master's degree with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University.