IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 22 November 2023, accepted 20 December 2023, date of publication 25 December 2023,
date of current version 3 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3346875

RESEARCH ARTICLE

Automatic Transportation Mode Classification Using a Deep Reinforcement Learning Approach With Smartphone Sensors

SIAVASH TAHERINAVID[1], SEYED VAHID MORAVVEJ[2], YEN-LIN CHEN[3], (Senior Member, IEEE), JING YANG[4], CHIN SOON KU[5], AND POR LIP YEE[4], (Senior Member, IEEE)

[1]School of Civil Engineering, Iran University of Science and Technology, Tehran 13114-16846, Iran
[2]Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan 84156-83111, Iran
[3]Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei 106344, Taiwan
[4]Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia
[5]Department of Computer Science, Universiti Tunku Abdul Rahman, Kampar 31900, Malaysia

Corresponding authors: Yen-Lin Chen (ylchen@mail.ntut.edu.tw), Por Lip Yee (porlip@um.edu.my), and Chin Soon Ku (kucs@utar.edu.my)

This work was supported in part by the National Science and Technology Council in Taiwan under Grant NSTC-112-2221-E-027-088-MY2 and Grant NSTC-111-2622-8-027-009; and in part by the Ministry of Education of Taiwan entitled "The study of artificial intelligence and advanced semiconductor manufacturing for female STEM talent education and industry-university value-added cooperation promotion", under Grant 1122302319 and Universiti Tunku Abdul Rahman (UTAR) Financial Support for Journal Paper Publication Scheme through UTAR, Malaysia.

ABSTRACT The increasing dependence on smartphones with advanced sensors has highlighted the imperative of precise transportation mode classification, pivotal for domains like health monitoring and urban planning. This research is motivated by the pressing demand to enhance transportation mode classification, leveraging the potential of smartphone sensors, notably the accelerometer, magnetometer, and gyroscope. In response to this challenge, we present a novel automated classification model rooted in deep reinforcement learning. Our model stands out for its innovative approach of harnessing enhanced features through artificial neural networks (ANNs) and visualizing the classification task as a structured series of decision-making events. Our model adopts an improved differential evolution (DE) algorithm for initializing weights, coupled with a specialized agent-environment relationship. Every correct classification earns the agent a reward, with additional emphasis on the accurate categorization of less frequent modes through a distinct reward strategy. The Upper Confidence Bound (UCB) technique is used for action selection, promoting deep-seated knowledge, and minimizing reliance on chance. A notable innovation in our work is the introduction of a cluster-centric mutation operation within the DE algorithm. This operation strategically identifies optimal clusters in the current DE population and forges potential solutions using a pioneering update mechanism. When assessed on the extensive HTC dataset, which includes 8311 hours of data gathered from 224 participants over two years. Noteworthy results spotlight an accuracy of 0.88 ± 0.03 and an F-measure of 0.87 ± 0.02, underscoring the efficacy of our approach for large-scale transportation mode classification tasks. This work introduces an innovative strategy in the realm of transportation mode classification, emphasizing both precision and reliability, addressing the pressing need for enhanced classification mechanisms in an ever-evolving digital landscape.

INDEX TERMS Transportation mode, sensor, smart phone, artificial neural network, reinforcement learning, differential evolution.

I. INTRODUCTION

Over the past few years, there has been a growing trend toward the increased popularity of smartphones. Usually,

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

a phone comes equipped with diverse sensors, including a GPS sensor, a gyroscope sensor, and a magnetometer, among others. As a result, obtaining a vast quantity of sensor data from smartphones is straightforward. In this paper, the data obtained from these sensors is used to recognize various modes of transportation. The classification of a person's mode of transportation is essential for the proper functioning of context-aware services. Leveraging the sensors integrated into smartphones has been acknowledged as a favorable method [1].

Many research studies have investigated this subject. Elhoushi et al. [2] proposed an algorithm that aims to recognize different indoor activities, including walking, standing, and various other movements. The input sensors used by the authors comprised a combination of accelerometer, gyroscope, and magnetometer triads, besides barometer data. Hemminki et al. [3] presented an algorithm that enables smartphones to identify five modes of transportation, namely car, bus, tram, train, and metro. Kinematic motion classifiers were used to differentiate between individuals who were walking and those who were not. After identifying motorized transportation, the classifier for motorized transport could determine the type of transportation activity being performed. Reddy et al. [4] used GPS and accelerometer data as input, which were subjected to noise removal, and established an instance-based decision tree as the classifier. They also used a discrete, hidden Markov model to make the final decision. Nham et al. [5] obtained accelerometer data and extracted features from it, such as the statistical properties of the signal and the magnitudes of 250 Fast Fourier Transform (FFT) components. The researchers used SVM and genetic data analysis to classify the data. Yu et al. [6] used the sensors for measuring angular velocity, linear acceleration, and magnetic field as input and extracted similar features from the data. There are two groups of methods for classifying transportation modes: location-based approaches [7] and sensor-based approaches [8], [9]. The initial type of classification relies on either GPS data or information from wireless networks [10], [11]. Unfortunately, methods that rely on location-based approaches may not work well in certain environments and can consume significant amounts of power [12], [13]. Yu et al. [6] stated that GPS and Wi-Fi consume a significant amount of power, 30 mA and 10.5 mA, respectively, which makes them impractical for portable devices. This article belongs to the class of sensor-based methods that do not rely on GPS and do not require significant power or resources [14], [15].

The impetus behind this article is to advance the state-of-the-art in transportation mode recognition by leveraging low-dimensional feature sets that enhance model efficiency in terms of size, processing speed, and accuracy, even when faced with overlapping signals. Addressing the challenges of data imbalance, our twofold contribution lies in not only improving accuracy but also in resource optimization. The current study evaluates performance against a substantial corpus of sensor data (over 1000 hours) spanning 10 modes of transportation.

Imbalanced data in transportation mode classification leads to overfitting and accuracy issues, as minority classes are less recognized. To mitigate this, data-level strategies like oversampling and undersampling seek to equalize class distribution but risk overfitting or loss of information [16]. Algorithmically, ensemble learning and cost-sensitive learning help by integrating multiple models and penalizing the misclassification of minority classes [17]. Using deep reinforcement learning (DRL) [18] presents an innovative avenue for addressing data imbalances, thanks to its unique capabilities. DRL implements a sophisticated reward structure that elevates the significance of underrepresented categories. It achieves this either by levying more substantial penalties for the incorrect classification of rare-class samples or by granting increased incentives for their correct identification. This strategy proactively mitigates the inherent inclination of traditional methods to favor the predominant category. The benefits of DRL are not limited to equalizing class proportions. It also enhances the detection of vital yet often marginalized patterns, particularly from less represented classes, through an efficient filtration of extraneous data. The prowess of DRL in extracting meaningful, albeit frequently ignored, attributes from the dataset plays a pivotal role in forging a model that is both resilient and effective [19].

To address inefficient feature extraction by traditional machine learning, which leads to poor generalization, protracted processing durations, and compromised precision [20], deep learning algorithms offer a potent alternative. Among these, the Multilayer Perceptron (MLP) stands out for its proficiency in identifying complex patterns and high-level features across diverse sectors. Initially devised to tackle the XOR dilemma, the MLP has been widely applied in pattern recognition and optimization tasks. It mimics the neural structure of the human brain by propagating input signals through a network of nodes spread across multiple layers. Each node aggregates incoming data and applies an activation function to produce an output signal. The distinct configuration of MLP allows for nodes within the same tier to remain unlinked, enhancing computational efficiency [19]. This algorithmic architecture helps to extract meaningful features from data, thus facilitating advanced machine learning applications.

To improve transportation mode classification, deep learning methods are pivotal, particularly for learning intricate data patterns. However, these methods face challenges such as getting stuck in local minima during optimization. This issue is critical, as it can lead to suboptimal performance [21]. To address this, Population-Based Metaheuristic (PBMH) algorithms like the Differential Evolution (DE) algorithm have been introduced. The DE algorithm is notable for avoiding local minima, thus enhancing optimization in multimodal scenarios. It is celebrated for its efficiency, effectiveness, and superior performance in weight initialization for deep learning models, which is crucial for the accuracy and speed of classification tasks [20], [22].

The DE algorithm comprises three primary steps: mutation, crossover, and selection. In mutation, a fresh potential solution arises by modifying the differences between current solutions. In the crossover phase, this newly created mutation vector merges with the initial vector. The selection process then determines the optimal solutions for the next cycle. The mutation function plays a crucial role in refining the efficiency of the DE algorithm. Its main goal is to allow the algorithm to scout and pinpoint new areas in the search landscape, preventing potential pitfalls like local optima. A meticulously crafted mutation function can amplify the algorithm's efficacy, promoting a broader and more varied search. Devising an efficient mutation function can be intricate and demanding, requiring a profound grasp of the optimization technique and the specific issue at hand [23]. Thorough evaluation and scrutiny are imperative during its formulation to ensure the mutation function is fine-tuned for optimal outcomes.

In this paper, we introduce an automated model for transportation mode categorization based on deep reinforcement learning. This model is distinct because of its inventive use of ANNs to enhance features and for envisioning the categorization task as a structured sequence of decision-making events. We incorporate an advanced DE algorithm for weight initialization and pair it with a unique agent-environment interaction. The model perceives categorization as a series of decisions, resembling a deduction game. Here, the agent is given a training sample to categorize via a defined policy. Agent effectiveness is gauged by rewards from correct or incorrect categorizations, offering higher incentives for accurate recognition of underrepresented classes. The goal is to maximize the aggregate reward by accurately categorizing the highest number of samples. We employ the UCB method for action choice to deepen knowledge and reduce dependency on randomness. A significant novelty is our cluster-focused mutation process within the DE algorithm. This process adeptly pinpoints the best clusters in the existing DE population, constructing potential solutions with a groundbreaking updated approach. Testing on the comprehensive HTC dataset, with 8311 hours of data from 224 participants over two years, revealed remarkable accuracy rates of 0.88 ± 0.03 and an F-measure of 0.87 ± 0.02. This highlights the potency of our strategy for extensive transportation mode categorization projects.

The article has several major contributions:

- We examine the classification of both transportation (including still, walk, run, bike, and vehicle) and vehicular (including motorcycle, car, bus, metro, train, and high-speed rail) modes. As far as we are aware, this is the first study to investigate these intricate features based on deep learning models.
- Our research aims to pinpoint features that efficiently classify transportation and vehicular modes under power and resource constraints. Prior studies, like those by Elhoushi et al. [2] and Figo et al. [24], focused on accuracy using an extensive feature set. In contrast, our benchmark, [6], identified seven low-power features for this purpose. Instead of creating new features, our goal is to judiciously select and amalgamate existing features that meet the limitations of power and size for classifying different modes of transport.
- · To address the challenge of imbalanced classification, we propose conceptualizing the prediction issue as a sequential decision-making task and introducing an algorithm based on reinforcement learning.
- The suggested structure uses the Upper Confidence Bound strategy for choosing actions, which can enhance the exploratory and exploitative aspects of the proposed methodological solution.
- We propose an encoding method that employs an improved DE algorithm to obtain the optimal initial value instead of randomly assigning weights.

The manuscript is organized in the following manner: Section two provides a detailed overview of the improved DE algorithm and its operational mechanisms. Section three unveils the model put forward by this study. In section four, the evaluation metrics, the dataset used, and a thorough examination of the findings are presented. The concluding section reflects on the study findings and suggests avenues for further investigation.

II. ENHANCED DIFFERENTIAL EVOLUTION

Differential Evolution (DE) [25] is a type of evolutionary algorithm used for optimization problems. It is a population-based optimization algorithm that aims to find the optimal solution for a function by iteratively updating a set of candidate solutions. DE works by creating new candidate solutions through a combination of information from the current population. This is achieved through a process of mutation, crossover, and selection. Mutation involves adding a scaled difference between two randomly selected candidate solutions to a third solution in the population to create a new solution. Crossover combines the newly created solution with an existing solution to generate a trial solution. Selection then determines whether the trial solution is better than the existing solution and updates the population accordingly. DE is a powerful and widely used optimization algorithm due to its simplicity, efficiency, and ability to solve a variety of optimization problems. It has been applied to various fields, including engineering, economics, and machine learning.

The mutation operator in DE plays a crucial role in the algorithm's performance, as it determines the diversity of candidate solutions in the population. If the mutation operator is too aggressive, it can lead to a loss of useful information in the population, resulting in premature convergence or stagnation. On the other hand, if the mutation operator is too weak, the population may not explore the search space effectively, resulting in slow convergence or suboptimal solutions. Several modifications have been proposed to address the sensitivity of DE to the mutation operator.

For example, researchers have proposed adaptive mutation schemes that adjust the mutation rate and scaling factor during the optimization process based on population performance. Other approaches include self-adaptive DE, where

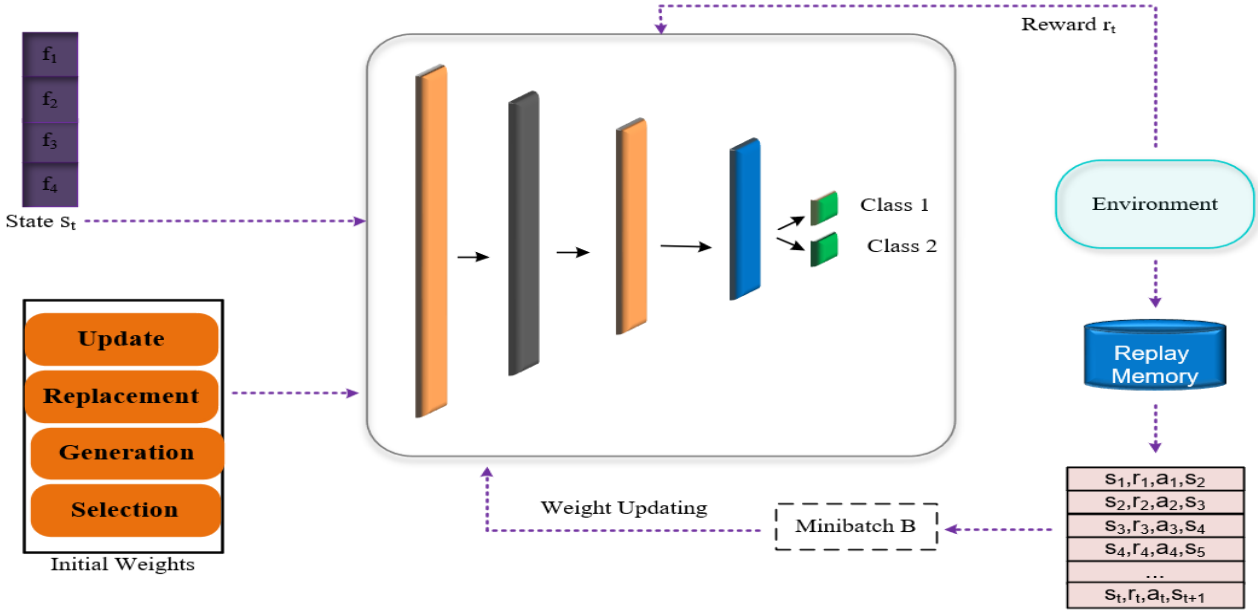


FIGURE 1. Overview of the proposed model: In Step 0, the model undergoes pre-training using the improved DE algorithm to initialize its weights. In Step 1, a sample S_t is drawn from the dataset (part of the RL environment) and fed to the network (Step 2). In Step 3, the action (the network prediction) is sent back to the environment to get the next sample s_{t+1} and reward rt at Step 4. The transition s_t,a_t,r_t,s_{t+1} is stored in the replay memory in Step 5. After storing multiple transitions in the replay memory, at Step 6, a mini-batch of transitions is drawn randomly to be used for updating the network weights in Step 7. The process is iterated until the network achieves an accurate classification of the input samples. The algorithm concludes when it exhausts the episodes.

the mutation rate and scaling factor are encoded as part of the candidate solutions and updated during the optimization process [23].

To enhance the performance of our DE algorithm, we use a mutation and updating technique based on clustering, aiming to improve the optimization process. Drawing inspiration from [26], the mutation operator proposed in this approach identifies a specific region within the search space. To divide the current population P into k clusters, encompassing different regions of the search space, the k-means algorithm is applied. The selection of the number of clusters is randomized within the range of $[2, \sqrt{N}]$. Following the clustering process, the cluster with the lowest mean fitness among its samples is determined to be the best cluster. The proposed mutation, based on clustering, is then defined as [27]:

$$\overrightarrow{v_i^{clu}} = \overrightarrow{win_g} + F(\overrightarrow{x_{r_1}} - \overrightarrow{x_{r_2}}) \tag{1}$$

where $\overrightarrow{x_{r_1}}$ and $\overrightarrow{x_{r_2}}$ represent a pair of randomly selected candidate solutions from the existing population, while $\overrightarrow{win_g}$ denotes the superior solution within the auspicious region. It is important to emphasize that $\overrightarrow{win_g}$ may not represent the optimal solution across the entire population.

After the creation of M novel solutions through cluster-oriented mutation, the population update proceeds under the Gradient-based Population Balance Algorithm (GPBA) [28], as follows [17]:

- **Selection**: Generate k random individuals to serve as the initial seeds for the algorithm.
- **Generation**: Produce a set of M solutions using mutation based on clustering and denote it as $\boldsymbol{v^{clu}}$.
- **Replacement**: Choose M solutions randomly from the current population to form set B.
- **Update**: Choose the best M solutions from the union of sets $\boldsymbol{v^{clu}}$ and B to form set B'. The new population is obtained by combining the elements of set P that are not in B with the elements of set $B'((P - B) \cup B')$.

III. PROPOSED METHOD

According to Fig. 1, the introduced model integrates MLP, DE, and RL, with each component's adoption justified as follows:

- **MLP:** MLP is adept at handling nonlinear challenges and boosts the precision of predictions and classifications by iteratively learning from its errors. The inherent architecture of layered nodes allows the MLP to discern and distinguish varied features judiciously, an essential capability for addressing the intricate and fluctuating datasets typically seen in healthcare applications.
- **RL:** To address the critical challenge of data imbalance, RL is deployed to steer the learning mechanism. It achieves this by particularly rewarding the correct identification of the less-representative class, ensuring that infrequent cases are effectively recognized and addressed.
- **DE:** DE is incorporated to scout an optimal initial position within the search landscape for the model back-propagation routine. This step is crucial for refining

the training phase and boosting the model's capacity to predict accurately.

In the following, we first describe the process of weight optimization using the proposed DE algorithm. Following this, we detail the training protocol of the MLP classifier through the application of RL techniques.

A. WEIGHT OPTIMIZATION

The proposed model contains a wide range of parameters, including those in MLP. In order to address this issue, we introduce a new method for parameter initialization using the improved DE algorithm. There are two key factors to consider in this approach: the encoding strategy and the fitness function. The encoding strategy defines the structure of each candidate solution, and the fitness function evaluates the quality of each candidate solution. By optimizing both aspects, we can effectively initialize the parameters in the model, leading to improved performance and accuracy.

B. ENCODING STRATEGY

To represent the weights of MLP in the proposed DE algorithm, a vector is created using the encoding approach. Because of the complex and non-linear nature of the MLP, obtaining precise weights for each neuron is a challenging task. Therefore, the encoding approach is used to organize the MLP weights into a vector that can be efficiently used by the DE algorithm. To determine the appropriate encoding approach, several iterations were performed, and the most suitable approach was chosen. An example of the encoding of an MLP with four hidden layers is shown in Fig. 2.

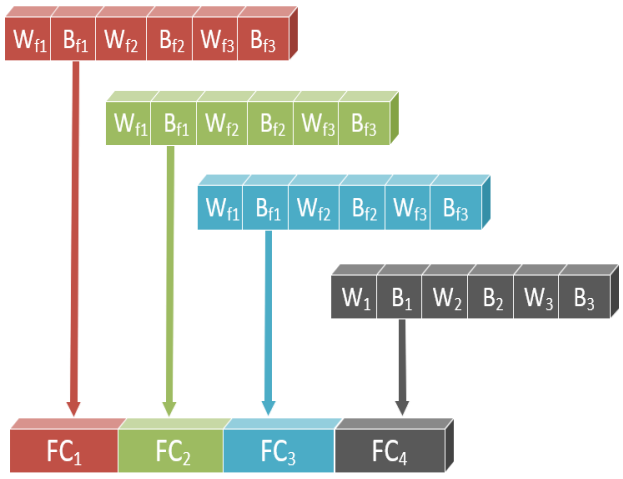


FIGURE 2. Illustration of weights in a vector. A large vector is created in which each element corresponds to a weight or bias.

C. FITNESS FUNCTION

The fitness function is a crucial component of the proposed model, as it evaluates the quality of each candidate solution generated by the DE algorithm. The fitness function is defined as a measure of the difference between the predicted output and the true output of the training instances. Specifically, the fitness function considers the sum of squared errors between the predicted output and the true output across all training instances. A lower fitness value indicates a better-quality solution, as it represents a smaller difference between the predicted output and the true output. By optimizing the fitness function, the proposed model can generate candidate solutions that provide accurate classifications for imbalanced datasets, ultimately leading to improved overall performance. In this research, we define our fitness function as:

$$Fitness = \frac{1}{\sum_{i=1}^{N} (y_i - \tilde{y}_i)^2} \tag{2}$$

where N signifies the aggregate count of training instances, with y_i and $\tilde{y}_i$ corresponding to the target and the predicted output of the i-th sample, respectively.

D. IMBALANCE CLASSIFICATION

One of the primary concerns in machine learning is dealing with imbalanced datasets, which is a common problem where the number of instances in one class is much higher than in the other. In such cases, the classifier output is biased towards the majority class, leading to poor performance for the minority class. This problem of class imbalance can significantly reduce the accuracy of a classification model, and it is challenging when one class has significantly more data than the other, as in our scenario, where most of the pairs are negative. Therefore, it is essential to develop effective algorithms that can handle imbalanced datasets to improve the classification performance of the model.

In this study, we propose a reinforcement learning approach to address imbalanced classification. Our approach views each sample in the training dataset as a state in the environment, with the neural network serving as the agent that makes a sequence of classification decisions on all samples. At each time step, the agent takes action by predicting the class label of the current sample, which serves as the state s_t. The environment then provides a reward r_t to guide the agent's next action. The reward function is carefully designed to assign higher values to correct classifications of the minority class and lower values to correct classifications of the majority class. This helps the agent prioritize learning to classify the minority class accurately, which is the main challenge in imbalanced classification problems.

$$r(s_t, a_t, l_t) = \begin{cases} +1, & a_t = l_t \text{ and } s_t \in D_P \\ -1, & a_t \neq l_t \text{ and } s_t \in D_P \\ \lambda, & a_t = l_t \text{ and } s_t \in D_N \\ -\lambda, & a_t \neq l_t \text{ and } s_t \in D_N \end{cases} \tag{3}$$

where D_N and D_P represent the majority and minority classes, respectively. A positive reward is given when a sample from the majority class is classified correctly, whereas a negative reward is assigned for an incorrect classification. Specifically, the reward for a correct classification is λ, where λ is a value between 0 and 1, while the reward for an incorrect classification is $-\lambda$. In the realm of deep Q-learning, the goal is for the agent to choose actions that will optimize the

cumulative future rewards, which are discounted by a factor of γ at each time step:

$$R_t = \sum_{t'=t}^{T} \gamma^{t'-t} r_{t'} \tag{4}$$

The formula represents the end of an episode at time-step T, which concludes when all samples have been classified or when a sample from the minority class is misclassified by the agent. The Q values refer to the quality of state-action pairs and are based on the expected return of the agent policy, π, given a state s and an action a:

$$Q^{\pi}(s, a) = E[R_t | s_t = s, a_t = a, \pi] \tag{5}$$

The maximum expected reward that can be obtained after taking action a and observing state s is represented by the optimal action-value function. This function considers all strategies:

$$Q^{*}(s, a) = max_{\pi} E[R_t | s_t = s, a_t = a, \pi] \tag{6}$$

The equation adheres to the principles of the Bellman equation, which posits that the optimal expected return for a chosen action is equal to the aggregate of the immediate rewards from the current action plus the highest expected return from subsequent actions at the following time step:

$$Q^{*}(s, a) = E[r + \gamma max_{a'} Q^{*}(s', a') | s_t = s, a_t = a] \tag{7}$$

The iterative update to estimate the optimal action-value function uses the Bellman equation:

$$Q_{i+1}(s, a) = E[r + \gamma max_{a'} Q_i(s', a') | s_t = s, a_t = a] \tag{8}$$

During the training process, the neural network generates an action a based on the state s. The environment then responds with a reward r and the subsequent state s'. These parameters are stored in a tuple (s, a, r, s') and saved in the replay memory, denoted as M. Furthermore, mini-batches B are drawn from the replay memory to perform gradient descent. The loss function is planned as follows:

$$\mathrm{L_i}(\theta_i) = \sum_{(\mathrm{s,a,r,s'}) \in \mathrm{B}} (\mathrm{y} - \mathrm{Q(s, a; \theta_i)})^2 \tag{9}$$

where θ denotes the model parameters, and y stands for the predicted target value of the Q function. The predicted target is equivalent to the immediate reward obtained from the current state-action pair, augmented by the discounted maximum expected future Q value:

$$y = r + \gamma max_{a'} Q(s', a'; \theta_{k-1}) \tag{10}$$

The Q value for the final state is zero. The gradient of the loss function in iteration i is computed as:

$$\nabla_{\theta_i} L(\theta_i) = -2 \sum_{(s,a,r,s') \in B} (y - Q(s, a; \theta_i)) \nabla_{\theta_i} Q(s, a; \theta_i) \tag{11}$$

Updating the model weights by minimizing the error can be achieved by performing a gradient descent step on the loss function:

$$\theta_{i+1} = \theta_i + \alpha \nabla_{\theta_i} Q(s, a; \theta_i) \tag{12}$$

where α signifies the learning rate of the neural network.

1) ACTION

In the realm of Q-learning, the ε-greedy technique is a widely adopted method to promote exploration. However, this technique has its shortcomings. It prioritizes immediate rewards from specific actions without adequately considering other vital factors that affect outcomes. As a result, the ε-greedy method may neglect to investigate actions that do not provide immediate benefits, especially in contexts characterized by limited data. Moreover, its dependence on random sampling makes it susceptible to variations because of a few initial negative results. With these limitations in mind, the ε-greedy algorithm may not be the ideal choice for our application, particularly in environments with limited data availability. To overcome these obstacles, we have fine-tuned our exploration strategy. This enhanced method emphasizes the depth of accumulated knowledge and reduces the reliance on random sampling. By utilizing the Upper Confidence Bound algorithm, we improve exploration efficiency. Here, choosing actions unfolds as follows:

$$a_t = arg \max_{a} [Q_t(a) + \varphi \sqrt{\frac{\ln n'}{n'_a}}] \tag{13}$$

where n' represents the aggregate count of choices made across all available actions, whereas n'_a signifies the number of times action a has been selected. The parameter $\varphi > 0$ determines the level or magnitude of exploration to be pursued.

2) OVERALL ALGORITHM

Our policy network structure has been intricately crafted to handle the complexity and size of our training data adeptly. Within this scheme, the configuration of the network inputs has been adjusted with precision to match the intricate details of our training set. Concurrently, the network outputs have been adjusted with great care to mirror the diverse classification labels present in our data. This refined approach is denoted as Algorithm 1.

It includes a detailed instruction set that guides the agent through a repetitive learning cycle until it reaches the optimal policy parameters. The method that underpins the decision-making for actions is founded on a self-directed heuristic, which guarantees that the agent's decisions are in harmony with its fundamental objectives. These decisions undergo evaluation based on the criteria delineated in Equation 3, furnishing a numerical measure of their impact and success. To bolster the resilience and effectiveness of our method, we deploy this protocol across E iterations. After concluding each iteration, we meticulously document the policy network parameters to meticulously chart the trajectory of the model learning process. Through this repetitive approach, we can methodically assess the performance of the model, laying the groundwork for strategic enhancements and precise refinement.

IV. EMPIRICAL EVALUATION

A. DATASET

The data used in this publication was obtained from the HTC Corporation and gathered over a two-year span starting

Algorithm 1 Pseudocode for the Proposed Model

Input: Dataset *Data* composed of $\{(x_1, y_1), (x_2, y_2), \ldots, (x_T, y_T)\}$, Maximum epoch E
Initialize network weights by pre-training network on the training data using the proposed DE algorithm
Initialize replay memory M and environment ε
for $i = 1 : E$
  Shuffle *Data*
  Set $s_1 = x_1$
  for $j = 1 : T$
    Select action a_t using Equation 13
    Determine reward r_t using Equation 3
    Set $s_{j+1} = x_{j+1}$
    Record the transition (s_j, a_j, r_j, s_{j+1}) into M
    Sample randomly a mini-batch of transition
(s_k, a_k, r_k, s_{k+1}) from M

$$y_k = \begin{cases} r_k, & \text{if } k+1 \textit{ is last step} \\ r_k + \gamma max_{a'} \hat{Q}\left(s_{k+1}, a'; \theta\right), & \textit{else} \end{cases}$$

    Accumulate gradients w.r.t θ:
    $\theta = \theta + \frac{\nabla L(\theta_k)}{\nabla(\theta_k)}$

TABLE 1. Description of the used dataset.

	Transportation Mode	Collection Time (h)
On a Vehicle	Still	158
	Walking	141
	Running	79
	Biking	98
	Motorcycle	163
	Car	208
	Bus	75
	Metro	132
	Train	89
	HSR	106

in 2012. The data collection involved 224 volunteers and amounted to 8311 hours, occupying 100 GB of storage [6]. The information used in this research was a subset of the unprocessed data from [5], which is approximately 20 GB in size and has been made available to the academic community by HTC. A diverse group of individuals participated in the study, with representation from different genders (60% male), body types, and age groups (ranging from 20 to 63 years old). There are ten modes included in the transportation state, namely: still, walk, run, bike, motorcycle, car, bus, metro, train, and high-speed rail (HSR). In contrast to other comparable research that uses small-scale data (only a few or a dozen hours) [29], the extensive data used in this study makes the findings more convincing and applicable.

Table 1 displays the database for five modes of transportation. In this paper, the vehicular modes (including motorcycle, car, bus, metro, train, and HSR) are grouped as a single mode referred to as ''on a vehicle''. Afterwards, the data will be divided into two sets: one for training and the other for testing to assess the model's performance.

TABLE 2. Sample sensor data for different modes of transportation.

Transportation Mode	Accelerometer (m/s^2) $[X, Y, Z]$	Gyroscope $(°/s)$ $[X, Y, Z]$	Magnetometer (μT) $[X, Y, Z]$
Still	[9.8,0.0, 0.0]	[0.0, 0.0, 0.0]	[23, 15, 10]
Walking	[9.5, 0.5, 0.3]	[0.2, 0.1, -0.1]	[24, 16, 9]
Running	[9.2, 1.2, 0.5]	[0.4, 0.3, -0.2]	[23, 14, 11]
Biking	[22, 17, 12]	[0.1, -0.1, 0.2]	[9.6, 0.4, -0.2]
Car	[9.7, 0.1, 0.1]	[0.1, 0.0, -0.1]	[24, 15, 11]
BUS	[9.81, 0.12, 0.15]	[0.002, -0.005, 0.01]	[23.2, 12.5, -49.7]
Metro	[9.77, 0.10, 0.18]	[0.003, 0.004, -0.01]	[24.1, 13.2, -48.8]
Train	[9.79, 0.13, 0.16]	[-0.002, 0.005, 0.01]	[23.8, 12.9, -49.2]

For the accelerometer, values represent acceleration due to movement on the respective axis. For the gyroscope, values represent angular velocities or the rate of change of the angular position of a rotation. For the magnetometer, values represent the strength and direction of the magnetic field.

Table 2 provides a comprehensive representation of sensor data from various transportation modes. The data presented is sampled from sensors measuring three different physical properties: acceleration, angular velocity, and magnetic field strength. The accelerometer measures the acceleration forces acting on an object, primarily because of movement and gravity. Its values are represented in the format [X, Y, Z], showcasing the acceleration on three distinct axes. For example, when still, the acceleration is close to gravitational pull, with minor variations because of subtle movements. Gyroscopes capture the rate of change in the angular position over time, often termed angular velocity. This information is vital for understanding the rotation or twist in the movement.

For instance, when walking, the slight angular velocities represented a hint of the foot's turning motion. Lastly,

magnetometers measure the magnitude and direction of the magnetic field surrounding them. This is useful for deducing orientation based on the Earth's magnetic field. The presented values in the magnetometer readings indicate the strength of the magnetic field at the respective axes. From the data, one can infer different movement patterns and the associated physical properties. For instance, the walking mode has slightly varied accelerometer readings compared with the still mode, indicating motion. The running mode, expectedly, shows higher variations due to increased pace and activity. Modes like biking and car travel show distinctive patterns influenced by the vehicle's movement. Modes of public transport, such as the bus, metro, and train, exhibit their unique signatures based on the vehicle's dynamics and the environment they operate in. Analyzing such data offers valuable insights into the underlying physics of each mode and aids in developing algorithms and systems that can accurately recognize and respond to these patterns.

This article uses a frame comprising 512 samples and generates the succeeding frame using a moving window approach with a 75% overlap. In this configuration, each frame is monitored for 17.06 seconds. The data continuity is smoothed, and the system delay is reduced by reusing 12.8 seconds of data as the next frame, which corresponds to a 75% overlap with the previous frame, in order to achieve a more consistent and continuous dataset. Note that we experimented with various overlap proportions, such as 50% and 90%. At 50%, we observed some minor inconsistencies in the dataset, leading to occasional misclassifications, especially during quick transitions between transportation modes. At 90%, while the results were slightly better than 50%, there was not a significant improvement from the 75% overlap, yet the computational costs were higher because of the increased redundancy.

Next, these frames will be converted into different characteristics. The seven features used in this study, based on the chosen baseline of [6], are provided as follows:

- The mean value of the magnitude of the accelerometer.
- The standard deviation of the magnitude of the accelerometer.
- The maximum value of the accelerometer's FFT.
- The accelerometer's highest FFT value is divided by its second-highest FFT value.
- The magnetometer's magnitude standard deviation is computed.
- The standard deviation of the gyroscope's measurement.
- The mean value of the gyroscope's measurement.

The aim of this research is not to introduce a novel statistical characteristic, as it has already been extensively explored. Rather than creating a new statistical feature, we endeavor to choose and merge helpful characteristics from prior research while considering the limitations of power and dimensions for tasks involving the classification of both transportation and vehicular modes. During the experiments conducted in this study, we explored numerous combinations of subsets using heuristic methods, and the most effective one was presented as the suggested feature. Subsequently, this article extracted six noteworthy characteristics using the previously mentioned features and Liu and Hsieh work [30] and subsequently identified eight additional features. The present study employs a combination of these 14 features to classify the training and testing data and assess its precision. It should be noted that the initial four features among those suggested were put forth in [6], whereas the fifth and sixth features were obtained from [30]. The subsequent features are explained as follows:

- The mean of the magnitude of the accelerometer.
- The magnitude's standard deviation of the accelerometer.
- The accelerometer's maximum FFT value.
- The mean value of the gyroscope.
- Comparison of acceleration on the z-axis with the gravitational force.
- Magnitude of the accelerometer in the X-Z plane, which is a horizontal section.
- The mean value of the acceleration on the X-axis.
- The mean value of the acceleration on the Y-axis.
- The mean value of the acceleration on the Z-axis.
- The highest value reached by the magnitude of the accelerometer.
- The mean value of acceleration that changes abruptly or instantaneously.
- The magnitude of variation in acceleration that changes abruptly or instantaneously.
- The mean value of the magnetometer readings.
- The mean value of a magnetic field that changes abruptly or instantaneously.

B. EXPERIMENTAL RESULTS

The model proposed in this study ran on a 64-bit Windows operating system equipped with 64 GB of RAM and a 64 GB Graphics Processing Unit (GPU). The training process was conducted for 119 epochs, which resulted in the development of the best model for the dataset. The total time taken for the training process was only 30 minutes, which indicates the computational efficiency of the model.

In this study, the proposed model was subjected to training and testing on a dataset alongside six other machine learning models: SVM [31], Naïve Bayes [32], KNN [33], Random forests [34], Logistic Regression [35], and Decision tree [36]. Two smaller versions of the proposed model were tested, namely Proposed + random weights and Proposed + random weights + RL. The performance of the algorithms was compared based on two sets of features: 7 features from [6] and 14 proposed features. The results were presented in two tables, Table 3 and Table 4. Overall, the results indicated that the proposed model outperformed all other models, including the Decision tree, which was the best competitor, in all criteria. The proposed model showed a significant reduction in error rates, specifically by over 62% and 38% in the two main criteria, F-measure and G-means. Comparison between the proposed model and Proposed + random weights and Proposed + random weights + RL revealed that the proposed model performance improved by approximately 80%, highlighting the effectiveness of the improved DE and RL

TABLE 3. Results of diverse classification techniques utilizing seven features from Reference [6] for transportation mode classification.

Algorithm	Accuracy	Recall	Precision	F-measure	G-means
SVM	0.54 ± 0.03	0.46 ± 0.03	0.42 ± 0.08	0.45 ± 0.03	0.52 ± 0.01
Naïve Bayes	0.65 ± 0.12	0.58 ± 0.00	0.60 ± 0.08	0.60 ± 0.08	0.64 ± 0.02
KNN	0.62 ± 0.07	0.53 ± 0.01	0.55 ± 0.04	0.54 ± 0.17	0.59 ± 0.01
Random forests	0.55 ± 0.08	0.45 ± 0.18	0.43 ± 0.07	0.44 ± 0.07	0.52 ± 0.20
Logistic Regression	0.65 ± 0.11	0.62 ± 0.03	0.57 ± 0.08	0.59 ± 0.11	0.64 ± 0.10
Decision tree	0.67 ± 0.11	0.66 ± 0.05	0.61 ± 0.13	0.63 ± 0.12	0.66 ± 0.01
Proposed + random weights	0.63 ± 0.02	0.64 ± 0.07	0.62 ± 0.00	0.62 ± 0.11	0.64 ± 0.20
Proposed + random weights + RL	0.67 ± 0.01	0.68 ± 0.09	0.66 ± 0.13	0.66 ± 0.02	0.67 ± 0.05
Proposed	0.70 ± 0.02	0.71 ± 0.07	0.68 ± 0.01	0.68 ± 0.01	0.71 ± 0.02

TABLE 4. Results of diverse classification techniques utilizing the proposed 14 features for transportation mode classification.

Algorithm	Accuracy	Recall	Precision	F-measure	G-means
SVM	0.69 ± 0.05	0.59 ± 0.07	0.54 ± 0.13	0.57 ± 0.07	0.66 ± 0.01
Naïve Bayes	0.82 ± 0.16	0.74 ± 0.00	0.77 ± 0.11	0.76 ± 0.11	0.81 ± 0.04
KNN	0.79 ± 0.10	0.67 ± 0.01	0.70 ± 0.05	0.69 ± 0.22	0.75 ± 0.01
Random forests	0.70 ± 0.11	0.57 ± 0.23	0.55 ± 0.10	0.56 ± 0.10	0.66 ± 0.26
Logistic Regression	0.82 ± 0.15	0.78 ± 0.05	0.72 ± 0.11	0.75 ± 0.15	0.81 ± 0.13
Decision tree	0.84 ± 0.15	0.83 ± 0.11	0.77 ± 0.27	0.80 ± 0.20	0.83 ± 0.01
Proposed + random weights	0.80 ± 0.05	0.81 ± 0.10	0.79 ± 0.00	0.78 ± 0.15	0.82 ± 0.25
Proposed + random weights + RL	0.85 ± 0.01	0.86 ± 0.12	0.84 ± 0.26	0.84 ± 0.04	0.85 ± 0.08
Proposed	0.88 ± 0.03	0.90 ± 0.10	0.87 ± 0.02	0.87 ± 0.02	0.90 ± 0.03

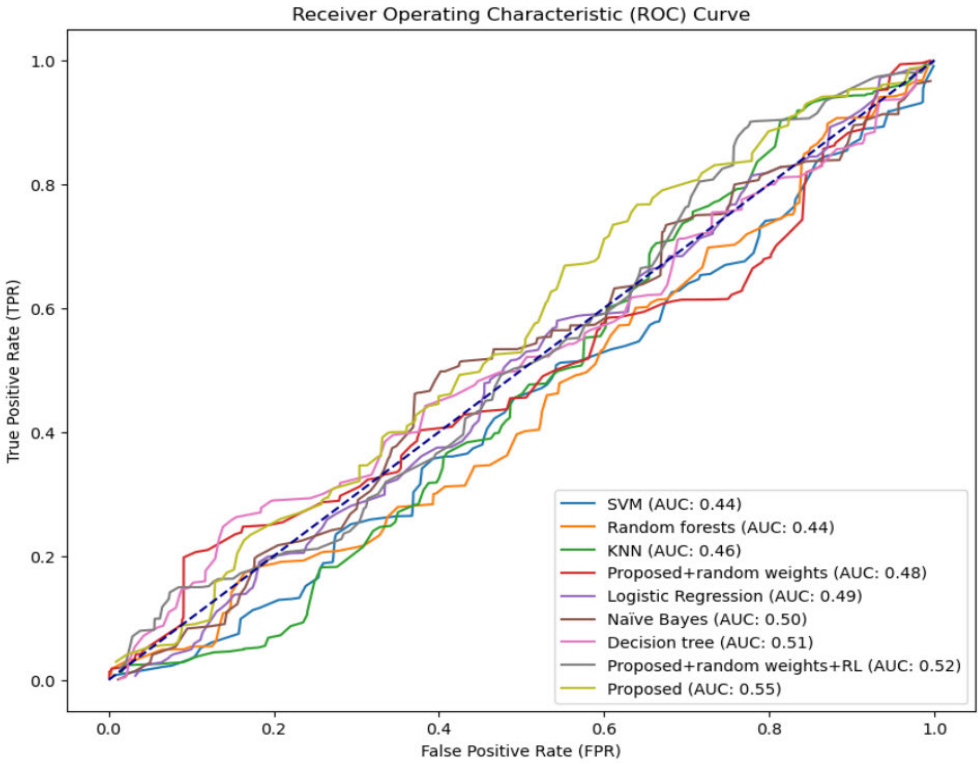
approaches. These findings suggest that the proposed model with the 14 features can yield more accurate and efficient results compared to other models tested in this study.

To discover the statistical significance of the results, we conducted a paired t-test between the proposed model and each of the other models for both feature sets. The results showed that the differences in performance metrics, including F-measure and G-means, were statistically significant ($p < 0.05$) for all comparisons. This indicates that the proposed model consistently outperformed or differed from the other models in terms of its performance metrics across both feature sets. The p-value of less than 0.05 suggests that the observed differences in F-measure and G-means are not because of random chance but a genuine reflection of the superiority or difference of the proposed model. Furthermore, using a paired t-test enabled us to account for potential correlations or relationships between the paired data points, thus giving a more accurate representation of the differences. The paired t-test assumes the data points being compared come in pairs, typically representing before-and-after measurements or two measurements under different conditions for the same subject. This type of analysis is beneficial when the data points are not independent, as it allows for the isolation of the effect of the model from other confounding variables.

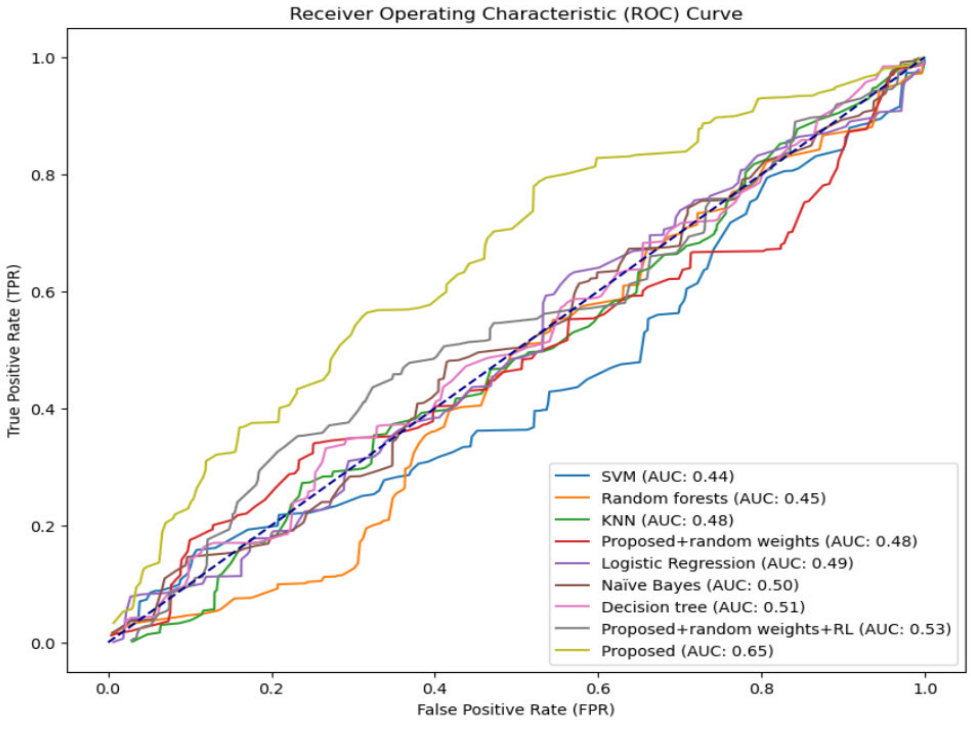
Fig. 3 showcases the receiver operating characteristic (ROC) curves for the methodologies identified in Tables 1 and 2. Our analysis uses the area under the curve (AUC), a singular metric quantifying a classifier's performance.

An AUC score of 1 represents flawless discrimination, while a score of 0.5 signifies discrimination indistinguishable from random guessing.

Our proposed method stood out from the crowd, achieving an impressive AUC of 0.55 (for 7 features) and 0.65 (for 14 features). This superior performance not only attests to its significant ability to accurately distinguish between positive and negative outcomes but also enhances our method's credibility as a proficient predictive tool. The Proposed + random weights + RL method attained an AUC of 0.52 (for 7 features) and 0.53 (for 14 features), emphasizing its robust capacity to differentiate between positive and negative cases. These results underscore the robustness and efficiency of our proposal, whether implemented individually or paired only with RL. Contrastingly, Decision Tree and Naïve Bayes reached only modest AUC values of 0.50 and 0.51, respectively. Despite their adequate discriminatory abilities, they did not match up to our proposed methods. Further down the ladder, SVM, Random Forests, and KNN showcased fewer promising performances, with AUC values ranging from 0.44 to 0.48. Notably, SVM lagged with a minimal AUC of 0.44, only marginally surpassing random chance. Our ROC analysis delineates the varied performances of the evaluated methods distinctly. The exceptional predictive capacity of our proposed method, employed alone or in combination with RL, exemplifies our approach's efficacy. This analysis not only lays the groundwork for potential enhancements but also heralds promising applications in the domain of prediction.



(a)



(b)

FIGURE 3. ROC diagram for the proposed model and other methods for transportation mode classification. The blue dashed line represents the ROC curve for a random guess. (a) ROC diagram for Table 3; (b) ROC diagram for Table 4.

C. VEHICLE MODE CLASSIFICATION

The different modes of transportation encompass HSR, metro, bus, car, and train. A comparative analysis between two sets of features is presented in Tables 4 and 5. The first set comprises seven features derived from [6] (Table 5), while the second set consists of the proposed features (Table 6). The findings demonstrated that the proposed model outperformed all other models, including the leading competitor, Decision

TABLE 5. Results of diverse classification techniques utilizing seven features from Reference [6] for vehicle mode classification.

Algorithm	Accuracy	Recall	Precision	F-measure	G-means
SVM	0.48 ± 0.13	0.39 ± 0.02	0.35 ± 0.09	0.40 ± 0.00	0.41 ± 0.02
Naïve Bayes	0.58 ± 0.02	0.52 ± 0.02	0.53 ± 0.18	0.51 ± 0.28	0.56 ± 0.06
KNN	0.53 ± 0.07	0.46 ± 0.00	0.48 ± 0.24	0.47 ± 0.21	0.50 ± 0.06
Random forests	0.48 ± 0.02	0.38 ± 0.02	0.36 ± 0.17	0.37 ± 0.20	0.44 ± 0.20
Logistic Regression	0.57 ± 0.21	0.54 ± 0.23	0.49 ± 0.18	0.52 ± 0.21	0.57 ± 0.20
Decision tree	0.60 ± 0.01	0.58 ± 0.55	0.53 ± 0.23	0.55 ± 0.02	0.55 ± 0.21
Proposed+random weights	0.56 ± 0.22	0.57 ± 0.17	0.54 ± 0.20	0.55 ± 0.01	0.57 ± 0.22
Proposed+random weights+FL	0.58 ± 0.21	0.60 ± 0.29	0.57 ± 0.02	0.58 ± 0.22	0.58 ± 0.15
Proposed	0.62 ± 0.52	0.63 ± 0.27	0.60 ± 0.21	0.62 ± 0.01	0.63 ± 0.22

TABLE 6. Results of diverse classification techniques utilizing the proposed 14 features for vehicle mode classification.

Algorithm	Accuracy	Recall	Precision	F-measure	G-means
SVM	0.61 ± 0.08	0.51 ± 0.07	0.46 ± 0.03	0.49 ± 0.08	0.58 ± 0.05
Naïve Bayes	0.74 ± 0.06	0.66 ± 0.10	0.69 ± 0.17	0.67 ± 0.24	0.73 ± 0.24
KNN	0.71 ± 0.12	0.59 ± 0.25	0.61 ± 0.15	0.60 ± 0.10	0.66 ± 0.11
Random forests	0.61 ± 0.10	0.49 ± 0.12	0.47 ± 0.25	0.48 ± 0.02	0.57 ± 0.06
Logistic Regression	0.74 ± 0.02	0.70 ± 0.25	0.64 ± 0.23	0.67 ± 0.02	0.74 ± 0.02
Decision tree	0.76 ± 0.10	0.75 ± 0.25	0.69 ± 0.17	0.72 ± 0.00	0.75 ± 0.21
Proposed+random weights	0.72 ± 0.25	0.73 ± 0.11	0.71 ± 0.20	0.72 ± 0.25	0.74 ± 0.12
Proposed+random weights+FL	0.77 ± 0.05	0.79 ± 0.22	0.76 ± 0.14	0.77 ± 0.14	0.78 ± 0.18
Proposed	0.81 ± 0.23	0.83 ± 0.20	0.84 ± 0.02	0.83 ± 0.12	0.83 ± 0.09

tree, across all evaluation criteria. The proposed model exhibited a substantial decrease in error rates, particularly by over 50% and 25% in the primary metrics of F-measure and G-means, respectively. A comparison between the proposed model and the variations incorporating random weights and reinforcement learning (Proposed + random weights, Proposed + random weights + RL) revealed an approximate 61% enhancement in the performance of the proposed model, emphasizing the effectiveness of the enhanced DE and RL methodologies.

Fig. 4 displays the ROC curves for the methodologies mentioned in Tables 4 and 5. The proposed method exhibited excellent performance, achieving AUC scores of 0.49 (for 7 features) and 0.57 (for 14 features), indicating its accurate discrimination between positive and negative outcomes. Similarly, the Proposed + random weights + RL method displayed a strong ability to differentiate with AUC scores of 0.48 (for 7 features) and 0.53 (for 14 features). SVM, Random Forests, and KNN showed fewer promising performances, with AUC values ranging from 0.43 to 0.47.

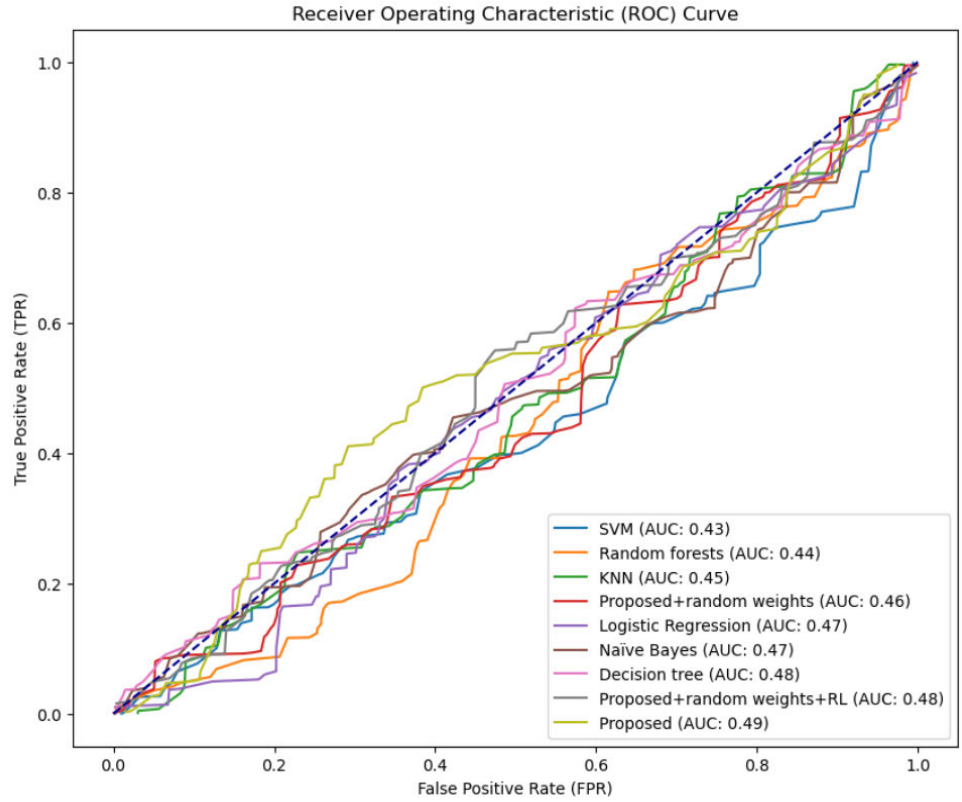
In Fig. 5 (a and b), the average value of the accelerometer's magnitude (the first original feature) and the standard deviation of the magnetometer's magnitude (the fifth original feature) are depicted, derived from a large database. To ensure a fair comparison, Fig. 5 (c and d) presents the added features measured using the same unit. Fig. 6 illustrates the average value of the sixth feature (representing the horizontal section in the X-Z plane of the accelerometer's magnitude) and the 14th added feature (average of magnetic instant change), respectively. This figure shows the supportive role of the added feature in the task, owing to its distinct properties. Including the additional feature leads to an enhancement in performing the vehicular mode detection task.

D. IMPACT OF OTHER METAHEURISTICS

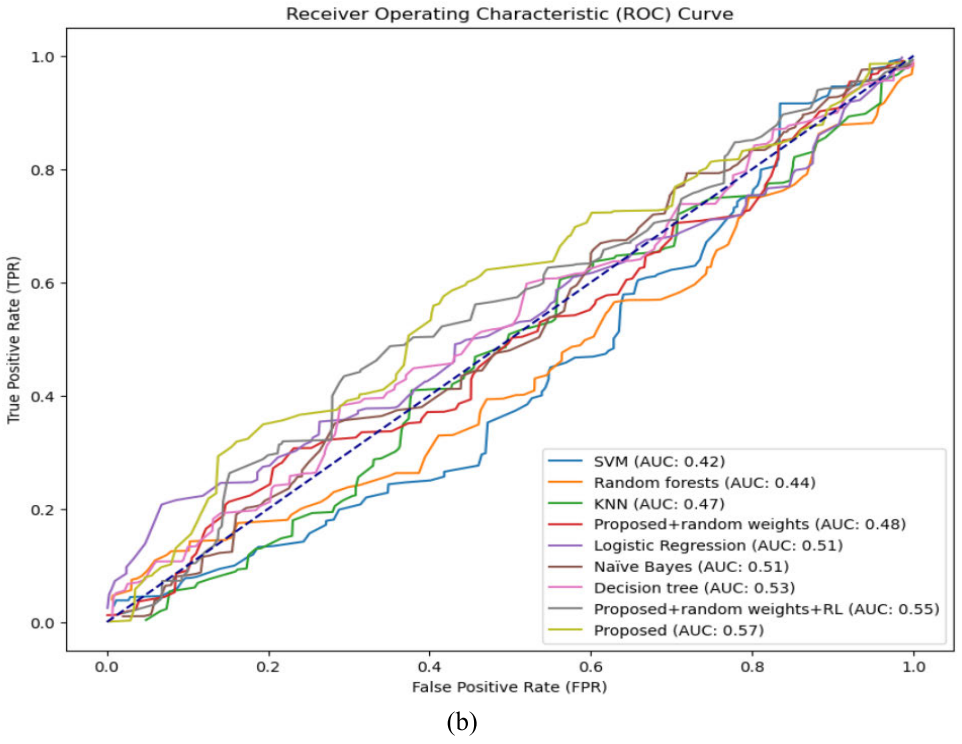
In the forthcoming experiment, our aim is to assess the efficacy of our enhanced DE algorithm compared to a range of metaheuristic optimization techniques. We aim to use diverse metaheuristics solely to determine the initial parameters of the model while maintaining all other aspects of the model unchanged. To achieve this, we have used six distinct algorithms, specifically standard DE, Artificial Bee Colony (ABC) [37], Firefly Algorithm (FA) [38], Bat Algorithm (BA) [39], Cuckoo Optimization Algorithm (COA) [40], and Grey Wolf Optimization (GWO) [41]. The results are presented in Table 6. As per the findings, the proposed DE technique showed a 53% reduction in error compared to the standard DE approach, indicating its superior ability. Moreover, the DE algorithm exhibited more desirable results than other algorithms, including ABC, GWO, and BA. These discoveries imply that the proposed approach surpasses existing state-of-the-art algorithms in terms of accuracy and resilience.

E. IMPACT OF THE REWARD FUNCTION

To assess the influence of the parameter λ on our model efficacy, we employed a nuanced reward mechanism that



(a)



(b)

FIGURE 4. ROC diagram for the proposed model and other methods for vehicle mode classification. The blue dashed line represents the ROC curve for a random guess. (a) ROC diagram for Table 4; (b) ROC diagram for Table 5.

attributes rewards and penalties of ± 1 and $\pm\lambda$ for accurate and erroneous identifications within the majority and minority categories, respectively. The magnitude of λ correlates with the prevalence ratio of these classes, optimally diminishing as the disparity in class representation widens. Our examination involved varying λ from 0 to 1 in 0.1 increments,

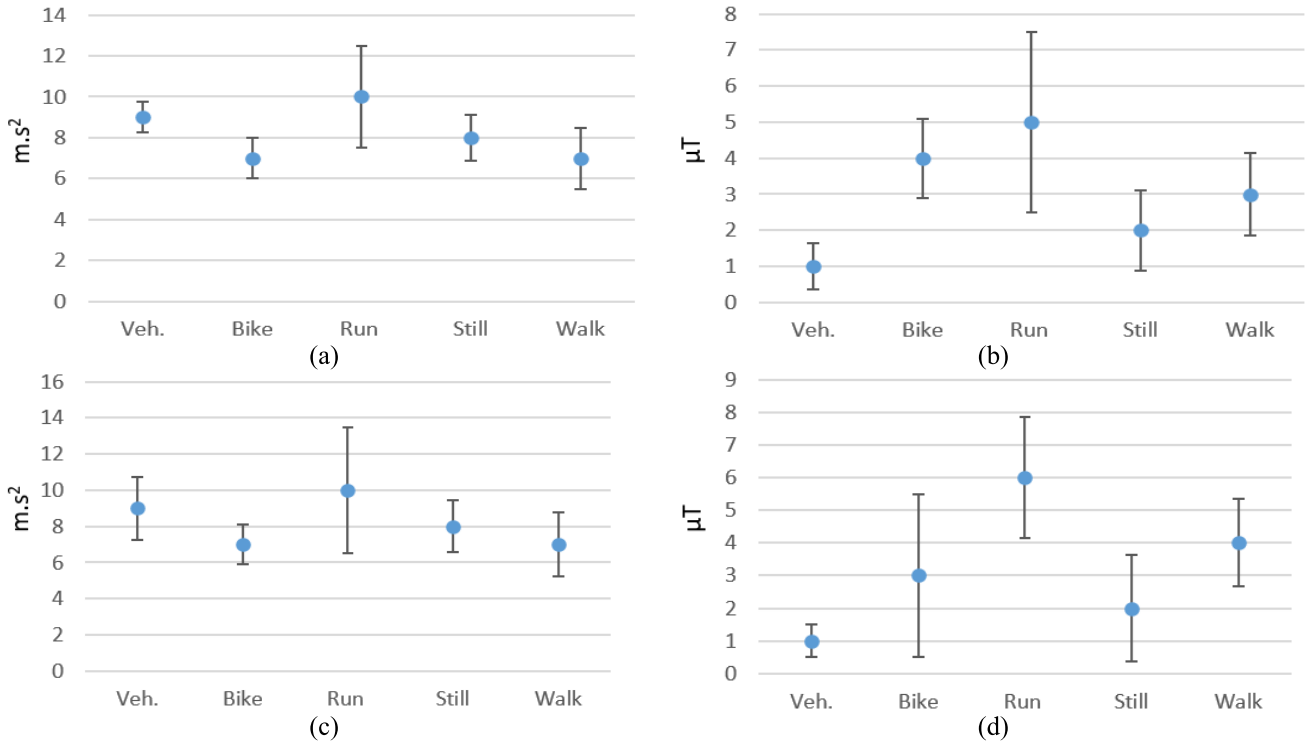


FIGURE 5. An example of an additional feature and its comparison with an original feature in transportation mode classification. (a) The initial feature derived from the average of the accelerometer's magnitude; (b) The fifth original feature obtained from the standard deviation of the magnetometer's magnitude; (c) The additional feature representing the horizontal section (X-Z plane) of the accelerometer's magnitude; (d) The 14th added feature calculated as the average of magnetic instant change.

TABLE 7. Results of various metaheuristic algorithms.

Algorithm	Accuracy	Recall	Precision	F-measure	G-means
DE	0.85 ± 0.15	0.84 ± 0.12	0.86 ± 0.05	0.84 ± 0.01	0.82 ± 0.41
ABC	0.84 ± 0.16	0.83 ± 0.10	0.84 ± 0.25	0.83 ± 0.02	0.80 ± 0.03
FA	0.83 ± 0.00	0.82 ± 0.12	0.81 ± 0.26	0.82 ± 0.07	0.78 ± 0.00
BA	0.81 ± 0.01	0.80 ± 0.01	0.80 ± 0.06	0.81 ± 0.14	0.76 ± 0.12
COA	0.79 ± 0.16	0.77 ± 0.01	0.79 ± 0.26	0.77 ± 0.17	0.74 ± 0.15
GWO	0.74 ± 0.14	0.72 ± 0.00	0.74 ± 0.00	0.73 ± 0.21	0.69 ± 0.11

with a consistent reward for correct majority class predictions, as illustrated in Fig. 7. At $\lambda = 0$, the model disregards the majority class influence, whereas at $\lambda = 1$, it gives equivalent weight to both classes. Optimal functionality was observed when λ was set to 0.4, as the model effectiveness rose with increases up to 0.4 and declined with further increments. Excessively minimizing λ might compromise the model's overall accuracy.

F. EXPLORING THE NUMBER OF MLP LAYERS

When designing a neural network, determining the optimal number of layers can be challenging. While increasing the number of layers may lead to a more powerful and expressive model, it also increases the risk of over-fitting the data, which can negatively impact its ability to generalize to unseen data. Too few layers may not provide enough flexibility to capture the key features of the training data, leading to poor performance. To find the optimal number of layers for the proposed method, we conducted experiments using different values of layers ranging from 1 to 12. We observed that performing the model initially improved as the number of layers increased from 1 to 4, but then declined for larger numbers of layers (see Fig. 8). These findings suggest that there is an optimal number of layers that balances model complexity and performance, and in this case, 4 was the most suitable number of layers.

G. IMPACT OF LOSS FUNCTION

To tackle imbalanced data, various conventional techniques can be employed, such as modifying data augmentation and loss functions. Among these techniques, the choice of loss function is more crucial, as it can help emphasize the minority

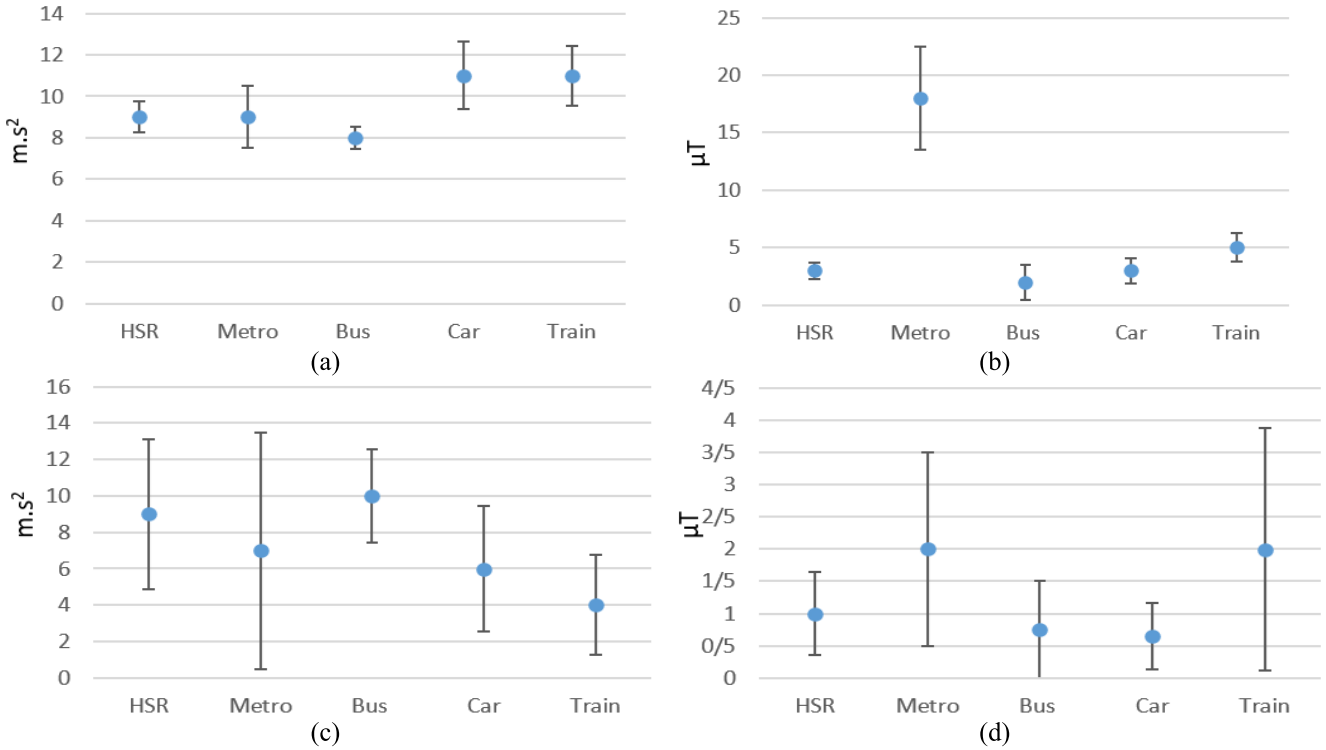


FIGURE 6. A representative additional feature and its comparison with an original feature in vehicular mode classification. (a) The initial feature derived from the average value of the accelerometer's magnitude; (b) The fifth original feature obtained by calculating the standard deviation of the magnetometer's magnitude; (c) The additional feature representing the horizontal section (X-Z plane) of the accelerometer's magnitude; (d) The 14th added feature calculated as the average value of the magnetic instant change.

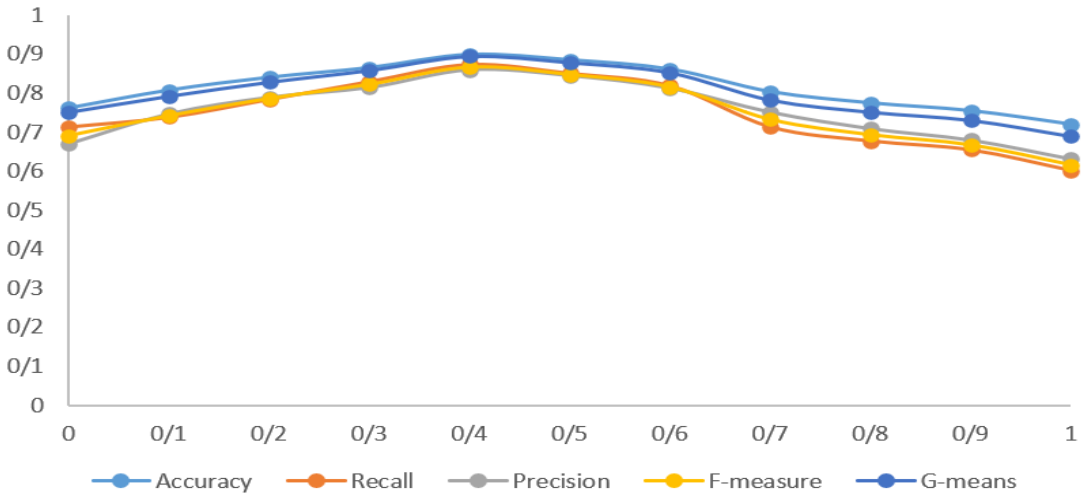


FIGURE 7. The proposed model performance metrics are plotted against the value of λ in the reward function.

class. In this study, we evaluated the effectiveness of five different loss functions, namely weighted cross-entropy (WCE) [42], balanced cross-entropy (BCE) [43], Dice loss (DL) [44], Tversky loss (TL) [19], and Combo Loss (CL) [45], on the proposed model. The BCE and WCE loss functions assign equal weight to both positive and negative samples. The CL function is useful for applications that have imbalanced data by giving less weight to simple examples and more weight to complex examples. As shown in Table 8, the CL function yields a lower error than TL, ranging from 26% to 44% for accuracy and F-measure metrics. However, performing the CL function is 61% worse than that of RL.

H. DISCUSSION

The findings of this study show the effectiveness of the proposed model for transportation mode detection using

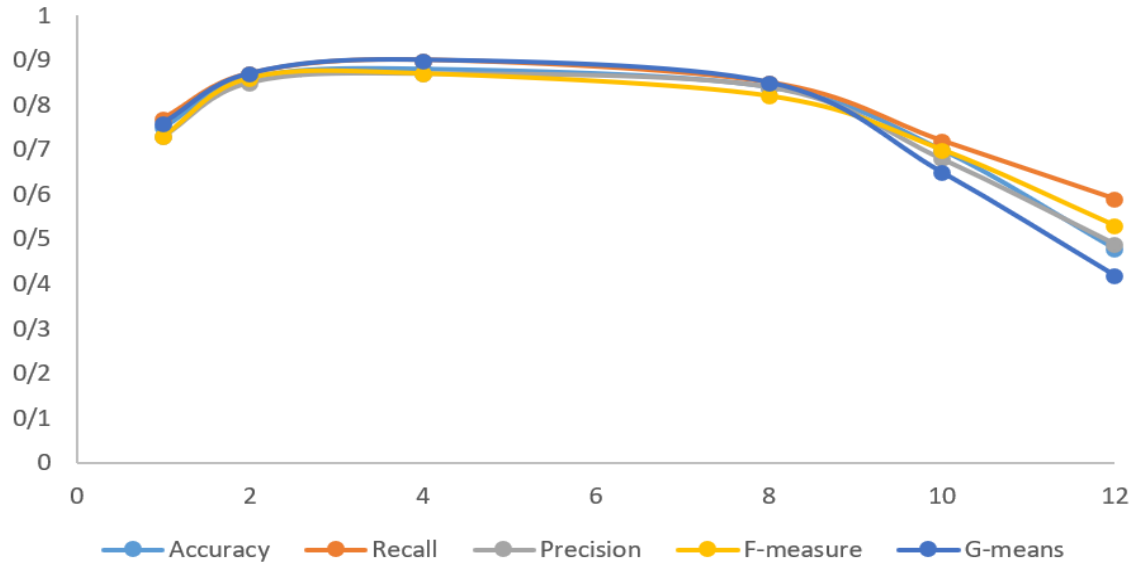


FIGURE 8. The performance metrics plotted vs. the different number of layers in MLP.

TABLE 8. Results of different loss functions.

Loss function	Accuracy	Recall	Precision	F-measure	G-means
WCE	0.75±0.03	0.74±0.01	0.73±0.12	0.74±0.00	0.76±0.03
BCE	0.80±0.02	0.79±0.05	0.77±0.16	0.77±0.01	0.81±0.00
DL	0.81±0.03	0.80±0.03	0.78±0.03	0.80±0.01	0.82±0.00
TL	0.83±0.12	0.82±0.01	0.80±0.02	0.81±0.04	0.84±0.06
CL	0.86±0.00	0.85±0.22	0.84±0.01	0.84±0.04	0.86±0.15

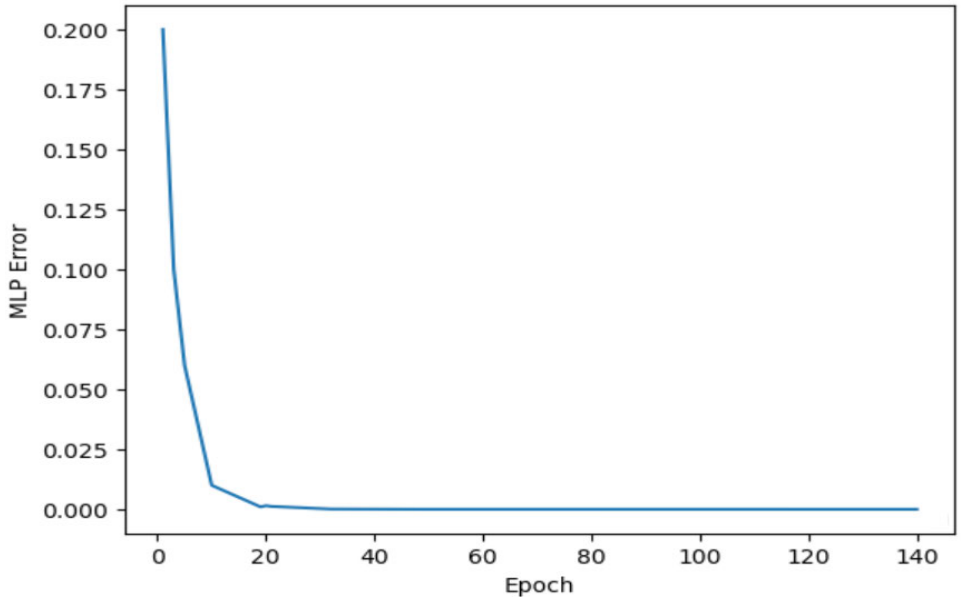


FIGURE 9. Diagram of the MLP error in classification.

smartphone sensor data. By employing an RL-based ANN and an enhanced DE algorithm, the model achieves superior results compared to other standard machine learning classifiers. Using two different feature vectors further enhances the accuracy of the model. Experimental evaluation of the dataset collected from the HTC company over two

years, involving 224 volunteers and a large amount of data, shows a high level of accuracy achieved by the suggested model. Compared to previous studies, this research makes several notable advancements. First, it examines the classification of both transportation and vehicular modes, which includes intricate features such as still, walk, run, bike, and various types of vehicles (motorcycle, car, bus, metro, train, and high-speed rail). To the best of our knowledge, this is the first study to investigate these features using deep learning models. Previous research has mainly focused on accuracy by utilizing numerous features, whereas this study aims to identify features that are suitable for both transportation and vehicular modes with limited power and resources. We analyzed the power consumption of various sensors and identified seven features for low-power transportation mode detection, which serves as a benchmark for this research. Interpreting the results reveals important insights into the classification of transportation modes using smartphone sensor data. The proposed RL-based ANN model, trained with the enhanced DE algorithm for weight initialization, shows significant improvements over traditional features and random weight initialization methods. By framing the prediction problem as a sequential decision-making process, the model effectively tackles imbalanced classification. The utilization of a specific reward system with a higher reward for identifying the minority class encourages the model to concentrate on detecting fraudulent transactions accurately.

The experiments conducted on the dataset provide valuable information regarding the optimal value for the reward function and the number of MLP layers, further optimizing the performance of the model. The error diagram of the MLP network is depicted in Fig. 9. This diagram effectively illustrates the training progress of the network over 140 epochs.

As the training advances, the error consistently decreases, indicating that the MLP network is improving its ability to classify automatic transportation modes accurately. The decreasing trend in error values signifies the convergence of the network towards an optimal solution. Notably, towards the conclusion of the training process, the error values reach an exceptionally low point, such as 0.0000001, underscoring the MLP network's high level of accuracy in automatic transportation mode classification.

The limitations of the proposed work are as follows:

- **Limited Generalizability**: The study primarily focuses on using data collected from the HTC company over a specific period involving a few volunteers. This raises concerns about the generalizability of the proposed model to other smartphone models, sensor configurations, and diverse user populations. The findings may not hold true for different datasets, which limits the applicability of the research in real-world scenarios.
- **Lack of Real-Time Evaluation**: The article does not discuss the performance of the proposed model in real-time settings. It would be valuable to know how the model performs when applied to real-time data streams, as transportation modes can transform. Without real-time evaluation, it is challenging to assess the practical feasibility and reliability of the model in dynamic environments.
- **Limited Discussion on Feature Selection**: While the article mentions selecting and combining useful features from existing studies, it lacks an in-depth discussion on the rationale behind feature selection. The importance and relevance of specific features in transportation mode detection remain unclear. Providing more insights into the feature selection process would enhance the transparency and reproducibility of the research.
- **Lack of Comparison with State-of-the-Art Approaches**: The article primarily compares the proposed model with traditional machine learning classifiers and metaheuristic initialization algorithms. However, it does not provide a comprehensive comparison with state-of-the-art approaches to transportation mode detection using smartphone sensor data. This omission limits the ability to assess the true novelty and superiority of the proposed model.
- **Scalability Challenges**: The study does not explicitly discuss the scalability of the proposed model to larger datasets or real-world implementation. As the volume of sensor data increases, the computational requirements and processing time may become a significant challenge. Considering scalability issues and discussing potential solutions would enhance the practicality and real-world applicability of the proposed model.

Finally, future works that can be considered are as follows:

- **Enhancing Transportation Mode Classification with Context-Aware Modeling:** It is important to consider the integration of special conditions into the classification model. Currently, variables such as weather and special vehicles like ambulances are not accounted for, which can significantly affect transportation mode classification. A promising direction would be to explore a context-aware approach to enhance model adaptability and accuracy under various real-world scenarios. For instance, drawing inspiration from studies like [46], we could develop a model that dynamically adjusts to the fluctuating conditions of the environment, providing a more holistic and precise classification performance.
- **Expansion to Diverse Smartphone Models and Sensor Configurations**: Future studies should indeed prioritize the validation of the proposed model across a more expansive array of smartphone models and sensor setups. This strategy would solidify the model's versatility, ensuring that it performs reliably on different devices, which is critical for its real-world application. Considering the variety of smartphone models and sensor configurations prevalent in the market, researchers can rigorously assess the resilience and efficacy of the model under a multitude of operational circumstances. As smartphones vary significantly in terms of hardware capabilities, sensor quality, and operating systems, these factors can influence the performance of sensor-based applications. For example, some sensors may have different levels of precision, sampling

rates, or noise profiles depending on the device's make and model. By testing the model on a wide range of devices, one can identify device-specific limitations or requirements, adapt the data preprocessing steps, and fine-tune the algorithm to account for such variations. Besides confirming the model's robustness, this comprehensive testing will help refine the model for better accuracy and reliability when deployed in a heterogeneous device environment, which is characteristic of the real world where end-users possess a diverse collection of smartphones.

- **Real-Time Evaluation and Implementation**: Conducting real-time assessments of the proposed model is indeed a crucial step in gauging its effectiveness in dynamic and time-sensitive contexts. By putting the model to the test in real-world scenarios as they unfold, researchers can critically evaluate its responsiveness, precision, and operational efficiency in identifying transportation modes at the moment they are being used. Implementing the model in real-time conditions offers a clear window into its applicability and practical utility for potential integration into systems like transport management and location-dependent services. Such real-world testing would help in identifying potential performance bottlenecks that might not be apparent in a simulated or controlled environment. It would also allow for the examination of the model's adaptability to sudden changes in transportation dynamics, such as shifts in traffic patterns or emergency situations, which require an agile response. By analyzing the model's real-time capabilities, developers can identify necessary improvements and optimizations, ensuring that the model not only performs well under laboratory conditions but also meets the rigorous demands of live environments. This includes assessing the computational demands of the model in real-time processing and the effectiveness of its decision-making process under the constraints of time and computational resources. Ultimately, real-time evaluations would inform developers and stakeholders about the model's readiness for deployment and provide concrete data to support its scalability for widespread use in intelligent transportation systems and smart city infrastructure.
- **Exploration of Different Reinforcement Learning Algorithms**: Although the current study puts forward a reinforcement learning-based algorithm tailored for imbalanced classification challenges, further research could pivot towards examining a spectrum of alternative reinforcement learning strategies. Undertaking comparative analyses and evaluations of disparate reinforcement learning algorithms within the realm of transportation mode detection would shed light on their comparative merits and overall performance metrics. Such an exploratory approach would help to pinpoint the algorithms that are most adept at managing imbalanced datasets, specifically within this domain. This future work could involve benchmarking standard algorithms like Q-learning, Deep Q Networks (DQNs), and Policy Gradient methods of more recent innovations like Proximal Policy Optimization (PPO) [47] and Soft Actor-Critic (SAC) [48] within the specific context of transportation mode detection. Researchers could assess these algorithms on various performance indicators, including their ability to balance precision and recall, handle skewed data distribution, and adapt to real-world sensor data. Furthermore, it would be beneficial to examine the impact of different reward structures, state representations, and the balance between exploration and exploitation on performing these algorithms in imbalanced classification tasks. Identifying the most effective algorithms and configurations could lead to more sophisticated models that are both accurate and efficient, offering substantial improvements over current systems for transportation mode detection and other applications where imbalanced datasets are prevalent.
- **Incorporation of Additional Sensor Data**: This study concentrates on harnessing data from three kinds of smartphone sensors: the accelerometer, magnetometer, and gyroscope. Prospective research endeavors could examine the incorporation of additional sensor data streams, for instance, GPS for location tracking, barometers for altitude estimation, or ambient light sensors for contextual environment information, to augment the precision and dependability of classifying transportation modes. The fusion of multiple sensor modalities holds the potential to enrich the model's capability to discern subtle characteristics and elevate its operational efficacy, especially in complex environments. For instance, GPS data could provide valuable insights into speed and routing patterns, while barometric pressure readings might offer cues about elevation changes that distinguish between modes of transport like walking, cycling, and driving on hilly terrain. Ambient light sensors could contribute contextual data that helps to determine the indoor or outdoor status of a user, adding another layer to the transportation mode inference process. The intricate interplay between these diverse sensors could yield a more holistic view of user movement and environment, reducing misclassification and improving the robustness of the model against sensor noise and variability in user behavior. Moreover, the exploration of advanced data fusion techniques and machine learning algorithms that can effectively combine these varied data sources would help to push the boundaries of what is achievable in transportation mode detection.
- **Optimization for Scalability**: With the burgeoning influx of sensor data, it is imperative to refine the proposed model to excel in scalability. Subsequent investigations should concentrate on forging methodologies that can adeptly manage expanding datasets with greater efficiency and curtailed computational demands. Optimizations for scalability might encompass strategies such as parallel processing to expedite computations, distributed computing to harness resources across

multiple machines, or implementing data reduction techniques that streamline the dataset while preserving essential information. By embracing these approaches, the model's efficiency could be significantly enhanced, facilitating its application on a more extensive scale. This would not only improve processing times but also enable the model to handle real-time data analysis, a critical factor for dynamic applications such as health monitoring systems, which require immediate feedback, and transport planning, where real-time data is essential for traffic management and predicting travel demand.

- **Integration with Other Applications**: The manuscript underscores the uses of transportation mode identification across several domains, such as health supervision, transit planning, and location-oriented services. Prospective studies could delve into amalgamating the suggested model within distinct applications pertinent to these fields. For instance, forging ahead with research into tailored health monitoring frameworks that incorporate transportation mode recognition presents an auspicious direction. This could involve devising systems that track and analyze individuals' transportation habits to offer insights into their physical activity levels and overall well-being. In transportation planning, such models could aid in optimizing traffic flow and public transit systems by providing detailed data on transportation patterns. In location-based services, enhanced transportation mode detection could refine recommendations and services provided to users based on their mode of travel.

V. CONCLUSION

The growing reliance on smartphones equipped with sophisticated sensors underscored the critical need for accurate transportation mode classification, essential for fields such as health monitoring and urban planning. This study was driven by the urgent requirement to improve transportation mode classification by capitalizing on the capabilities of smartphone sensors, especially accelerometers, magnetometers, and gyroscopes. In addressing this need, we developed a novel automated classification model based on deep reinforcement learning. Our model distinguished itself through a unique method of extracting advanced features using ANNs and conceptualizing the classification task as a well-organized sequence of decision-making steps. We refined the DE algorithm for weight initialization and integrated it with a specialized agent-environment relationship. Each successful classification provided the agent with a reward, focusing especially on the precise identification of infrequent modes via an adaptive reward strategy. We employed the UCB method for action selection, enhancing profound learning and reducing dependency on random factors. A significant advancement in our research was the creation of a cluster-focused mutation process within the DE algorithm. This process adeptly pinpointed superior clusters within the existing DE population and generated promising solutions through a novel update technique. Evaluated against the comprehensive HTC dataset, which comprised 8311 hours of data collected from 224 individuals over two years, our model achieved remarkable results, demonstrating an accuracy of 0.88 ± 0.03 and an F-measure of 0.87 ± 0.02, highlighting the method's effectiveness for extensive transportation mode classification tasks.

From a practical perspective, the findings of this work have significant implications for transportation planning and management. The ability to accurately detect transportation modes using smartphone sensors can be pivotal for smart cities, aiding in traffic flow predictions, transportation planning, and optimization of public transport services. Furthermore, our method can be integrated into mobile applications to provide users with real-time data on their movement, offering personalized route suggestions or even health-related advice based on detected activity. This could lead to a more informed, efficient, and environmentally friendly transportation system, reducing congestion and pollution while enhancing the user experience.

One potential direction in the future could be to expand the dataset to include more diverse transportation modes and environmental conditions to assess the model's generalization ability. Additionally, investigating the applicability of the proposed method to other similar problems, such as pedestrian activity recognition or vehicle type classification, could provide insight into its versatility. Another avenue could be to explore the use of other reinforcement learning techniques, such as deep Q-networks or policy gradient methods, to handle data imbalances and improve model performance further. Finally, evaluating the proposed method's performance in real-world scenarios, such as integrating it into a transportation management system or a mobile application, could provide valuable feedback on its practical usability and impact.

REFERENCES

[1] M. Gjoreski, V. Janko, G. Slapničar, M. Mlakar, N. Reščič, J. Bizjak, V. Drobnič, M. Marinko, N. Mlakar, M. Luštrek, and M. Gams, "Classical and deep learning methods for recognizing human activities and modes of transportation with smartphone sensors," *Inf. Fusion*, vol. 62, pp. 47–62, Oct. 2020, doi: 10.1016/j.inffus.2020.04.004.

[2] M. Elhoushi, J. Georgy, A. Noureldin, and M. J. Korenberg, "Motion mode recognition for indoor pedestrian navigation using portable devices," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 1, pp. 208–221, Jan. 2016, doi: 10.1109/TIM.2015.2477159.

[3] S. Hemminki, P. Nurmi, and S. Tarkoma, "Accelerometer-based transportation mode detection on smartphones," in *Proc. 11th ACM Conf. Embedded Networked Sensor Syst.* New York, NY, USA: ACM, Nov. 2013, pp. 1–14, doi: 10.1145/2517351.2517367.

[4] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava, "Using mobile phones to determine transportation modes," *ACM Trans. Sensor Netw.*, vol. 6, no. 2, pp. 1–27, Feb. 2010, doi: 10.1145/1689239.1689243.

[5] B. Nham, K. Siangliulue, and S. Yeung, "Predicting mode of transport from iPhone accelerometer data," Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep., 2011. [Online]. Available: https://cs229.stanford.edu/proj2008/NhamSiangliulueYeung-PredictingModeOfTransportFromIphoneAccelerometerData.pdf

[6] M.-C. Yu, T. Yu, S.-C. Wang, C.-J. Lin, and E. Y. Chang, "Big data small footprint: The design of a low-power classifier for detecting transportation modes," *Proc. VLDB Endowment*, vol. 7, no. 13, pp. 1429–1440, Aug. 2014, doi: 10.14778/2733004.2733015.

[7] L. Stenneth, O. Wolfson, P. S. Yu, and B. Xu, "Transportation mode detection using mobile phones and GIS information," in *Proc. 19th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.* New York, NY, USA: ACM, Nov. 2011, pp. 54–63, doi: 10.1145/2093973.2093982.

[8] S. Wang, C. Chen, and J. Ma, "Accelerometer based transportation mode recognition on mobile phones," in *Proc. Asia–Pacific Conf. Wearable Comput. Syst.* Shenzhen, China: IEEE Press, Apr. 2010, pp. 44–46, doi: 10.1109/APWCS.2010.18.

[9] J. Lester, T. Choudhury, and G. Borriello, "A practical approach to recognizing physical activities," in *Pervasive Computing* (Lecture Notes in Computer Science, Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Berlin, Germany: Springer, vol. 3968, 2006, pp. 1–16, doi: 10.1007/11748625_1.

[10] P. Widhalm, P. Nitsche, and N. Brändie, "Transport mode detection with realistic smartphone sensor data," in *Proc. 21st Int. Conf. Pattern Recognit. (ICPR)*, Nov. 2012, pp. 573–576.

[11] D. Ashbrook and T. Starner, "Using GPS to learn significant locations and predict movement across multiple users," *Pers. Ubiquitous Comput.*, vol. 7, no. 5, pp. 275–286, Oct. 2003, doi: 10.1007/s00779-003-0240-0.

[12] Y. Zheng, L. Liu, L. Wang, and X. Xie, "Learning transportation mode from raw gps data for geographic applications on the Web," in *Proc. 17th Int. Conf. World Wide Web*. New York, NY, USA: ACM, Apr. 2008, pp. 247–256, doi: 10.1145/1367497.1367532.

[13] Y. Zheng, Y. Chen, Q. Li, X. Xie, and W.-Y. Ma, "Understanding transportation modes based on GPS data for Web applications," *ACM Trans. Web*, vol. 4, no. 1, pp. 1–36, Jan. 2010, doi: 10.1145/1658373.1658374.

[14] L. Liao, D. J. Patterson, D. Fox, and H. Kautz, "Building personal maps from GPS data," *Ann. New York Acad. Sci.*, vol. 1093, no. 1, pp. 249–265, Dec. 2006, doi: 10.1196/annals.1382.017.

[15] S. Nath, "ACE: Exploiting correlation for energy-efficient and continuous context sensing," in *Proc. 10th Int. Conf. Mobile Syst., Appl., Services*. New York, NY, USA: ACM, Jun. 2012, pp. 29–42, doi: 10.1145/2307636.2307640.

[16] M. Soleimani, Z. Forouzanfar, M. Soltani, and M. Jafari Harandi, "Imbalanced multiclass medical data classification based on learning automata and neural network," *EAI Endorsed Trans. AI Robot.*, vol. 2, Jul. 2023, doi: 10.4108/airo.3526.

[17] S. V. Moravvej, S. J. Mousavirad, M. H. Moghadam, and M. Saadatmand, "An LSTM-based plagiarism detection via attention mechanism and a population-based approach for pre-training parameters with imbalanced classes," in *Neural Information Processing* (Lecture Notes in Computer Science, Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Cham, Switzerland: Springer, vol. 13110, 2021, pp. 690–701, doi: 10.1007/978-3-030-92238-2_57.

[18] Z. Yao, X. Liang, G.-P. Jiang, and J. Yao, "Model-based reinforcement learning control of electrohydraulic position servo systems," *IEEE/ASME Trans. Mechatronics*, vol. 28, no. 3, pp. 1446–1455, Jun. 2023, doi: 10.1109/TMECH.2022.3219115.

[19] S. Zhang, C. Tjortjis, X. Zeng, H. Qiao, I. Buchan, and J. Keane, "Comparing data mining methods with logistic regression in childhood obesity prediction," *Inf. Syst. Frontiers*, vol. 11, no. 4, pp. 449–460, Sep. 2009, doi: 10.1007/s10796-009-9157-0.

[20] S. V. Moravvej, R. Alizadehsani, S. Khanam, Z. Sobhaninia, A. Shoeibi, F. Khozeimeh, Z. A. Sani, R.-S. Tan, A. Khosravi, S. Nahavandi, N. A. Kadri, M. M. Azizan, N. Arunkumar, and U. R. Acharya, "RLMD-PA: A reinforcement learning-based myocarditis diagnosis combined with a population-based algorithm for pretraining weights," *Contrast Media Mol. Imag.*, vol. 2022, pp. 1–15, Jun. 2022, doi: 10.1155/2022/8733632.

[21] M. Soltani, M. Bahadori, and M. Soleimani, *Optimal Predictive Maintenance and Spare Part Inventory Policies for a Degrading System Subjected to Imperfect Actions*. Accessed: Sep. 20, 2023.[Online]. Available: https://ssrn.com/abstract=4558570.

[22] M. Soltani, J. P. Kharoufeh, and A. Khademi, "Optimal call center staffing and pricing under QoS constraints," in *Proc. Annu. Conf. Proc., New Orleans, LA: Inst. Ind. Syst. Eng. (IISE)*, 2023, pp. 1–6.

[23] S. V. Moravvej, S. J. Mousavirad, D. Oliva, G. Schaefer, and Z. Sobhaninia, "An improved DE algorithm to optimise the learning process of a BERT-based plagiarism detection model," in *Proc. IEEE Congr. Evol. Comput. (CEC)*. Padua, Italy: IEEE Press, Jul. 2022, pp. 1–7, doi: 10.1109/CEC55065.2022.9870280.

[24] D. Figo, P. C. Diniz, D. R. Ferreira, and J. M. P. Cardoso, "Preprocessing techniques for context recognition from accelerometer data," *Pers. Ubiquitous Comput.*, vol. 14, no. 7, pp. 645–662, Oct. 2010, doi: 10.1007/s00779-010-0293-9.

[25] K. V. Price, "Differential evolution," in *Handbook of Optimization: From Classical to Modern Approach* (Intelligent Systems Reference Library), Berlin, Germany: Springer, vol. 38, 2013, pp. 187–214, doi: 10.1007/978-3-642-30504-7_8.

[26] S. J. Mousavirad and H. Ebrahimpour-Komleh, "Human mental search: A new population-based metaheuristic optimization algorithm," *Int. J. Speech Technol.*, vol. 47, no. 3, pp. 850–887, Oct. 2017, doi: 10.1007/s10489-017-0903-6.

[27] S. Vahid Moravvej, S. Jalaleddin Mousavirad, D. Oliva, and F. Mohammadi, "A novel plagiarism detection approach combining BERT-based word embedding, attention-based LSTMs and an improved differential evolution algorithm," 2023, *arXiv:2305.02374*.

[28] K. Deb, "A population-based algorithm-generator for real-parameter optimization," *Soft Comput.*, vol. 9, no. 4, pp. 236–253, Apr. 2005, doi: 10.1007/s00500-004-0377-4.

[29] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *Pervasive Computing* (Lecture Notes in Computer Science, Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Berlin, Germany: Springer, vol. 3001, 2004, pp. 1–17, doi: 10.1007/978-3-540-24646-6_1.

[30] C.-H. Liu and S. Hsieh, "A fall detection system using accelerometer and gyroscope," M.S. thesis, Dept. Comput. Sci. Eng., Tatung Univ., Taipei, Taiwan, 2011.

[31] H. Wang and D. Hu, "Comparison of SVM and LS-SVM for regression," in *Proc. Int. Conf. Neural Netw. Brain*. Beijing, China: IEEE press, 2005, pp. 279–283, doi: 10.1109/ICNNB.2005.1614615.

[32] G. I. Webb, E. Keogh, and R. Miikkulainen, "Naïve Bayes," in *Encyclopedia of Machine Learning*. Boston, MA, USA: Springer U.S., 2011, pp. 713–714, doi: 10.1007/978-0-387-30164-8_576.

[33] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "KNN model-based approach in classification," in *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE* (Lecture Notes in Computer Science, Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Berlin, Germany: Springer, vol. 2888, 2003, pp. 986–996, doi: 10.1007/978-3-540-39964-3_62.

[34] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[35] M. P. LaValley, "Logistic regression," *Circulation*, vol. 117, no. 18, pp. 2395–2399, May 2008, doi: 10.1161/CIRCULATIONAHA.106.682658.

[36] A. J. Myles, R. N. Feudale, Y. Liu, N. A. Woody, and S. D. Brown, "An introduction to decision tree modeling," *J. Chemometrics*, vol. 18, no. 6, pp. 275–285, Jun. 2004, doi: 10.1002/cem.873.

[37] S. Vakilian, S. V. Moravvej, and A. Fanian, "Using the artificial bee colony (ABC) algorithm in collaboration with the fog nodes in the Internet of Things three-layer architecture," in *Proc. 29th Iranian Conf. Electr. Eng. (ICEE)*, Tehran, Iran, May 2021, pp. 509–513, doi: 10.1109/ICEE52715.2021.9544399.

[38] N. F. Johari, A. M. Zain, M. H. Noorfa, and A. Udin, "Firefly algorithm for optimization problem," *Appl. Mech. Mater.*, vol. 421, pp. 512–517, Sep. 2013, doi: 10.4028/www.scientific.net/AMM.421.512.

[39] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," *Nature Inspired Cooperat. Strategies Optim.*, vol. 284, pp. 65–74, Oct. 2010, doi: 10.1007/978-3-642-12538-6_6.

[40] S. Vakilian, S. V. Moravvej, and A. Fanian, "Using the cuckoo algorithm to optimizing the response time and energy consumption cost of fog nodes by considering collaboration in the fog layer," in *Proc. 5th Int. Conf. Internet Things Appl. (IoT)*. Isfahan, Iran: IEEE press, May 2021, pp. 1–5, doi: 10.1109/IoT52625.2021.9469722.

[41] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[42] Ö. Özdemir and E. B. Sönmez, "Weighted cross-entropy for unbalanced data with application on COVID X-ray images," in *Proc. Innov. Intell. Syst. Appl. Conf. (ASYU)*. Istanbul, Turkey: IEEE press, Oct. 2020, pp. 1–6, doi: 10.1109/ASYU50717.2020.9259848.

[43] F. Huang, J. Li, and X. Zhu, "Balanced symmetric cross entropy for large scale imbalanced and noisy data," 2020, *arXiv:2007.01618*.

[44] X. Li, X. Sun, Y. Meng, J. Liang, F. Wu, and J. Li, "Dice loss for data-imbalanced NLP tasks," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 465–476.

[45] S. Danaei, A. Bostani, S. V. Moravvej, F. Mohammadi, R. Alizadehsani, A. Shoeibi, H. Alinejad-Rokny, and S. Nahavandi, "Myocarditis diagnosis: A method using mutual learning-based ABC and reinforcement learning," in *Proc. IEEE 22nd Int. Symp. Comput. Intell. Informat. 8th IEEE Int. Conf. Recent Achievements Mechatronics, Autom., Comput. Sci. Robot. (CINTI-MACRo)*, Budapest, Hungary: IEEE press, Nov. 2022, pp. 265–270, doi: 10.1109/CINTI-MACRo57952.2022.10029403.

[46] F. Kiani and Ö. F. Saraç, "A novel intelligent traffic recovery model for emergency vehicles based on context-aware reinforcement learning," *Inf. Sci.*, vol. 619, pp. 288–309, Jan. 2023, doi: 10.1016/j.ins.2022.11.057.

[47] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2023, *arXiv.1707.06347*.

[48] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, and S. Levine, "Soft actor-critic algorithms and applications," 2018, *arXiv:1812.05905*.

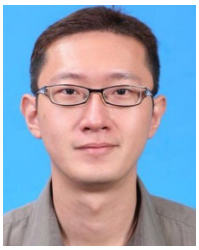
SIAVASH TAHERINAVID received the bachelor's degree from the Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran, in 2019, and the master's degree from the Iran University of Science and Technology, in 2022. He is currently a Research Assistant with the Transportation Data Analytics Center (TDAC), Iran University of Science and Technology. His research interests include applications of data science and big data in travel demand analysis and forecasting, deep learning algorithms (machine learning), cognitive driving and autonomous vehicles, human behavior and choice-making in transportation, naturalistic driving, intelligent transportation systems (ITS), and sustainable public transportation design.

SEYED VAHID MORAVVEJ received the bachelor's degree from Kashan University, Isfahan, Kashan, Iran, in 2017, and the master's degree from the Isfahan University of Technology, Isfahan, in 2020. He is currently a Researcher with the Department of Computer Science at the Isfahan University of Technology. His research interests include deep learning and natural language processing.

YEN-LIN CHEN (Senior Member, IEEE) received the B.S. and Ph.D. degrees in electrical and control engineering from National Chiao Tung University, Hsinchu, Taiwan, in 2000 and 2006, respectively. From February 2007 to July 2009, he was an Assistant Professor with the Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan. From August 2009 to January 2012, he was an Assistant Professor with the Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei, Taiwan, where he was an Associate Professor, from February 2012 to July 2015. Since August 2015, he has been a Full Professor with the National Taipei University of Technology. His research results have been published on over 100 journals and conference papers. His research interests include artificial intelligence, intelligent image analytics, embedded systems, pattern recognition, intelligent vehicles, and intelligent transportation systems. He is a fellow of the IET and a member of ACM, IAPR, and IEICE.

JING YANG received the Bachelor of Engineering degree in navigation technology from Shandong Jiaotong University, in 2022. He is currently pursuing the master's degree in data science with Universiti Malaya. His primary research interests include medical image processing and deep learning.

CHIN SOON KU received the Ph.D. degree from Universiti Malaya, Malaysia, in 2019. He is currently an Assistant Professor with the Department of Computer Science, Universiti Tunku Abdul Rahman, Malaysia. His research interests include AI techniques, such as genetic algorithm, computer vision, decision support tools, graphical authentication (authentication, picture-based password, and graphical password), machine learning, deep learning, speech processing, natural language processing, and unmanned logistics fleets.



POR LIP YEE (Senior Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees from Universiti Malaya, Malaysia. He is currently an Associate Professor with the Faculty of Computer Science and Information Technology, Universiti Malaya. His research interests include machine learning, support vector machines, deep learning, long-short-term memory, computer vision, the AIoT, the IoT, blockchain, authentication, graphic passwords, PIN-entry, cryptography, data hiding, stenography, and watermarking.