sustainability

MDPI

Article

Improved Artificial Rabbits Optimization with Ensemble Learning-Based Traffic Flow Monitoring on Intelligent Transportation System

Mahmoud Ragab [1,2,*], Hesham A. Abdushkour [3], Louai Maghrabi [4], Dheyaaldin Alsalman [5], Ayman G. Fayoumi [6] and Abdullah AL-Malaise AL-Ghamdi [6,7]

1 Information Technology Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia
2 Mathematics Department, Faculty of Science, Al-Azhar University, Naser City, Cairo 11884, Egypt
3 Nautical Science Department, Faculty of Maritime Studies, King Abdulaziz University, Jeddah 21589, Saudi Arabia; habdushakour@kau.edu.sa
4 Department of Software Engineering, College of Engineering, University of Business and Technology, Jeddah 21448, Saudi Arabia; l.maghrabi@ubt.edu.sa
5 Department of Cybersecurity, School of Engineering, Computing and Design, Dar Al-Hekma University, Jeddah 34801, Saudi Arabia; dsalman@dah.edu.sa
6 Information Systems Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia; afayoumi@kau.edu.sa (A.G.F.); aalmalaise@kau.edu.sa (A.A.-M.A.-G.)
7 Information Systems Department, School of Engineering, Computing and Design, Dar Al-Hekma University, Jeddah 34801, Saudi Arabia
* Correspondence: mragab@kau.edu.sa

Citation: Ragab, M.; Abdushkour, H.A.; Maghrabi, L.; Alsalman, D.; Fayoumi, A.G.; AL-Ghamdi, A.A.-M. Improved Artificial Rabbits Optimization with Ensemble Learning-Based Traffic Flow Monitoring on Intelligent Transportation System. *Sustainability* **2023**, *15*, 12601. https://doi.org/10.3390/su151612601

Academic Editors: Keshuang Tang, Hong Zhu and Ashish Bhaskar

Received: 23 June 2023
Revised: 29 July 2023
Accepted: 8 August 2023
Published: 20 August 2023
Corrected: 3 June 2025

Abstract: Traffic flow monitoring plays a crucial role in Intelligent Transportation Systems (ITS) by dealing with real-time data on traffic situations and allowing effectual traffic management and optimization. A typical approach used for traffic flow monitoring frequently depends on collection and analysis of the data through a manual process that is not only resource-intensive, but also a time-consuming process. Recently, Artificial Intelligence (AI) approaches like ensemble learning demonstrate promising outcomes in numerous ITS applications. With this stimulus, the current study proposes an Improved Artificial Rabbits Optimization with Ensemble Learning-based Traffic Flow Monitoring System (IAROEL-TFMS) for ITS. The primary intention of the proposed IAROEL-TFMS technique is to employ the feature subset selection process with optimal ensemble learning so as to predict the traffic flow. In order to accomplish this, the IAROEL-TFMS technique initially designs the IARO-based feature selection approach to elect a set of features. In addition, the traffic flow is predicted using the ensemble model that comprises a Gated Recurrent Unit (GRU), Long Short-term Memory (LSTM), and Bidirectional Gated Recurrent Unit (BiGRU). Finally, the Grasshopper Optimization Algorithm (GOA) is applied for the adjustment of the optimum hyperparameters of all three DL models. In order to highlight the improved prediction results of the proposed IAROEL-TFMS algorithm, an extensive range of simulations was conducted. The simulation outcomes imply the supremacy of the IAROEL-TFMS methodology over other existing approaches with a minimum RMSE of 16.4539.

Keywords: Intelligent Transportation System; traffic flow; prediction models; deep learning; artificial rabbits optimizer

1. Introduction

A transportation system is a massive and sophisticated system that is strongly associated with the day-to-day life activities of human beings. Intelligent Transport Systems (ITSs) play a significant part in leading the future approaches for transportation [1]. The

area of research on ITSs is large and a developing one while its connotations are regularly modernized via the consolidated and advanced fields like computer technologies, information science, and advanced transportation theories. ITSs are intelligent systems that deploy innovative technologies for modeling transportation systems and the regulation of traffic flow. Thus, it offers the end users with huge information and protection, in addition to the qualitative improvement of the interaction levels between the transport users, unlike the traditional transportation systems [2]. The ITSs comprise the entire transportation services of the cities containing ambulances, fire departments, traffic police, and so on [3]. The most vital mechanisms behind these systems involve two digital entities such as the application of mathematical modeling approaches for analyzing the transport network and the development of proposals to resolve the transportation issues such as pedestrian flows and optimization of traffic, traffic management, public transport, investment justification in the construction of transport infrastructure and the optimization of traffic lights [4]. Integrating the technologies with ITSs for strategic transportation planning results in the proficient usage of the current transportation infrastructures and select the correct path for its advancement in the future [5].

The reliance on traffic flows depends upon real-time traffic information along with the recorded data collected from various types of sensor sources such as cameras, radars, inductive loops, mobile Global Positioning Systems (GPS), social sites, and crowdsourcing [6]. Traffic data is exploding on a day-to-day basis since a huge volume of data gets generated in conventional sensors and recent technologies [7]. In recent years, transportation control and management has become a data-driven domain. Though numerous traffic flow prediction models and systems were used earlier, most of the models were shallow traffic models and failed even in the case of a large number of dimensional datasets. Traffic Flow Prediction (TFP) is categorized into long-term TFP, medium-term TFP, and short-term TFP based on time [8]. Both medium-term and long-term prediction measures are commonly based on years, months, weeks, and days. Because of the larger time, the data stabilities are comparatively effective and so it can be generally utilized for prediction. However, the short-term TFP is usually done at a time range of 5–15 min. Owing to this short time, the data stability is comparatively insufficient while the data is highly-sophisticated and the random variation is enormous. This scenario increases the complexity of the prediction works [9]. In this background, it is challenging to attain highly accurate performance in real-time traffic data detection due to the increased complexity of traffic situations and the development of highly accurate short-term TFP [10]. As an alternative, deep learning (DL) methods engage several researchers and manufacturers, because of their capability to manage the motion models' classification complexity, understand the natural languages, reduce the dimensionality, and detect the objects. The DL techniques utilize multiple layers of neural network approaches in order to gain the essential characteristics from low to higher levels of information. Thus, it can recognize huge volumes of structures in the datasets, which ultimately support imagining and creating purposeful interpretations from the information.

The current study presents an Improved Artificial Rabbit Optimization with an Ensemble Learning-based Traffic Flow Monitoring System (IAROEL-TFMS) for the ITSs. The IAROEL-TFMS technique initially designs the IARO-based feature selection approach to elect a set of features. In addition, the TFP process is executed using an ensemble model, comprising a Gated Recurrent Unit (GRU), Long Short-term Memory (LSTM), and Bidirectional Gated Recurrent Unit (BiGRU). Finally, the Grasshopper Optimization Algorithm (GOA) is applied for optimum hyperparameter adjustment of all three DL models. In order to highlight the improved prediction outcomes of the IAROEL-TFMS system, an extensive range of simulations was conducted and the outcomes were evaluated.

2. Related Works

Djenouri et al. [11] developed a jointed graph optimizer and a predictor in a single pipeline, based on the convolution graph-based NN approach, in order to predict the

urban traffic flow in edge IoT environments. First, preprocessing was executed to get rid of the noise from the sequence of original road network data collected from urban traffic. Then, the outlier detection technique was implemented to effectively remove the noise and other irrelevant patterns after which the road network data was explored further. The new optimization method was designed for fine-tuning the hyperparameter values of the given framework. Han and Huang [12] introduced a short-term TFP method based on the DL approach. At first, road network data compression was introduced based on CX decomposition and correlation study methods. Next, the traffic flow data was randomly split into fluctuation terms and trend terms with the help of the spectral decomposition technique. Then, the impact of the trend term on traffic prediction was removed. The authors in [13] devised a logistic agent-based technique to analyze the data from public transportation like trains, cars, and buses. This intelligent logistic design was constructed on a parallel NN model called Swarm-NN (SWNN). This technique recognizes public transportation and analyzes the sensory data.

Qi et al. [14] presented a DL architecture based on Federated learning and Asynchronous GCN for accurate prediction of traffic flow on a real-time basis. The presented technique exploits the asynchronous spatial–temporal GCN to design the spatial–temporal dependency upon traffic data. Cheng et al. [15] proposed a short-term TFP method. Many common DL algorithms and other shallow prediction methods were considered to compare with the proposed method. The results demonstrate that the outcome of the proposed model was higher than the rest of the techniques in predicting short-term TFP. Next, the multi-feature speed prediction for the spatial location was carried out using the CNN-LSTM module. Chan et al. [16] aimed at overcoming the problems in traffic flow prediction by proposing three strategies: (i) a missing data handling system using a weighted historical information system called Weighted Missing Data Imputation (WEMDI); (ii) simulation of the live traffic models; and (iii) pheromone-based NN for traffic prediction along with rerouting method. The authors in [17] developed a cloud–vehicle–road framework that describes the exact message content along with message generation and broadcast processes. In this framework, the binary classifier technique was employed to evaluate the uploaded traffic-related message so as to enhance the performance. Ma et al. [18] developed a hybrid spatial–temporal FS approach (STFSA) involving the CNN-GRU technique. Firstly, the STFSA approach was followed to reconstruct the spatiotemporal matrices of traffic speed based on spatial and temporal features. Next, the non-linear fitting capability of the CNN was used for the extraction of deep features in both pooling and convolutional layers for the training models.

Neelakandan et al. [19] developed an Optimum-stacked Sparse Auto-encoder-based TFP (OSSAE-TFP) approach for the ITS. The purpose of the OSSAE-TFP approach was to define the traffic flow level in the ITS. Furthermore, the SSAE-based forecast method was planned for TFP whereas better hyperparameters of the SSAE approach were altered with the deployment of Water Wave Optimizer (WWO). Liang et al. [20] presented a spatio–temporal multi-GCN (STMGCN)-based vessel TFP approach by utilizing various kinds of inherent correlations in the created maritime graph. Xia et al. [21] examined a short-term TFP approach that integrates community detection-based federated learning and GCN in order to overcome the issues like time-consuming training methods, superior communication costs, and data privacy threats faced by global GCNs, since the volume of data increases on a daily basis. The federated community GCN (FCGCN) accomplished accurate, timely, and safe traffic forecasts based on big traffic data, which was vital for the effectual functioning of ITS.

In [22], a Deep Hybrid Attention (DHA) technique comprising both traffic and weather blocks was developed. The traffic block contained the Convolutional Neural Network (CNN) and the Gated Recurrent Unit (GRU) neural network in order to capture the spatio–temporal rules of traffic flow data. Next, the weather block made use of the Convolutional Long Short-term Memory (ConvLSTM) model to derive the relationship between weather and traffic flow data. In this model, a self-attention mechanism was integrated

between the blocks. In [23], the authors developed a cloud-assisted ITS and traffic control system. The cloud server executes the traffic data and controls the efficiency of the signals. In addition to these, the alerts are notified to the closer control room at the time of traffic congestion. In [24], the authors proposed a deep encoder–decoder predictive model based on variational Bayesian inference. The Bayesian neural network was developed by integrating variational inference with GRU. Moreover, the variational inference was applied to the multi-head attention mechanism in order to eliminate noise-induced deterioration of the predictive results. In [25], an effective model named RPConvformer was presented in which the enhanced components include 1D causal convolutional sequence embedding and relative position encoding.

3. The Proposed Model

In the current research work, the authors have designed a novel IAROEL-TFMS methodology for traffic flow monitoring. The major aim of the proposed IAROEL-TFMS technique is to employ the feature subset selection process with optimal ensemble learning in order to forecast the traffic flow. To accomplish this, the IAROEL-TFMS technique comprises three stages of operations, namely, IARO-based feature subset selection, ensemble classification, and GOA-based hyperparameter tuning. Figure 1 depicts the workflow of the proposed IAROEL-TFMS method.

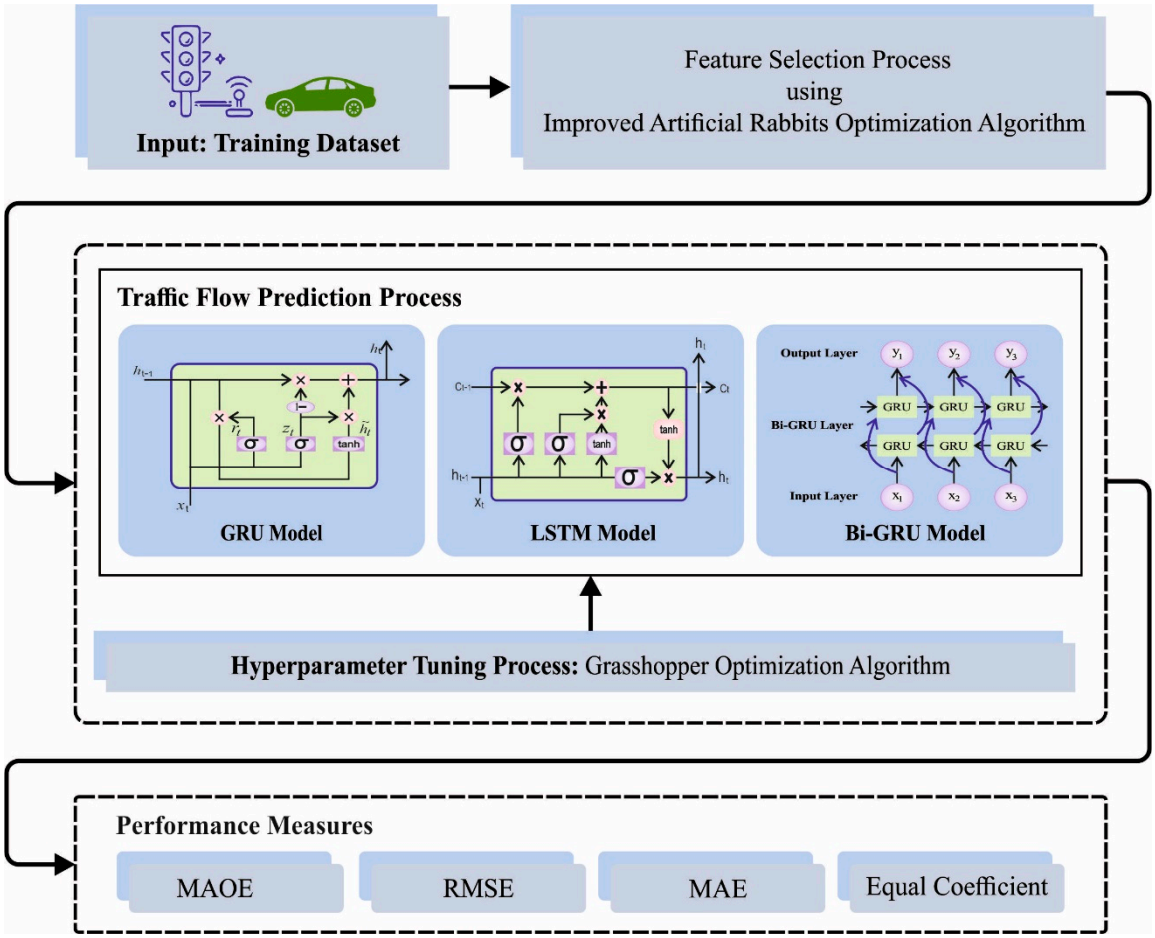

Figure 1. Workflow of the IAROEL-TFMS algorithm.

3.1. Feature Selection using IARO Algorithm

For optimal selection of the features, the IARO algorithm is employed. The ARO technique [26] is presented based on the survival strategies followed by the rabbits. It involves three search strategies such as the detour foraging strategy, random hiding strategy, and

energy shrink strategy. The detour foraging strategy involves the exploration; the random hiding strategy is dedicated to exploitation; and the energy shrink strategy enhances the balance between the exploration and exploitation phases.

Consider that all the rabbits in the population have their territory, whereas every region has d burrows and edible grass and the individual rabbit often visits the foraging area in an arbitrary fashion. The mathematical modeling of the detour foraging strategy is given herewith.

$$v_i(z+1) = xj(z) + R\cdot(x_i(z) - xj(z)) + round(0.5\cdot(0.05 + r_1))\cdot n_1, \quad i = 1, \cdots, n, j \neq i \tag{1}$$

$$R = L\cdot C \tag{2}$$

$$L = \left(e - e^{\left(\frac{z-1}{Z}\right)^2}\right)\cdot\sin(2\pi r_2) \tag{3}$$

$$c(k) = \begin{cases} 1 \ if \ k = g(l) \\ 0 \ else \end{cases} \quad k = 1, \cdots, d \text{ and } l = 1, \cdots, \left\lceil r_3\frac{z-1}{Z}d \right\rceil \tag{4}$$

$$g = randperm\ (d) \tag{5}$$

$$n_1 \sim N(0,1). \tag{6}$$

Let $v_j(z+1)$ be the candidate place of the ith-rabbit at z+1 time and c indicates the mapping vector. r_1, r_2, and r_3 are three random numbers between (0, 1). $x_j(z)$ indicates the location of the ith rabbit at t time, n demonstrates the number of rabbits, d represents the dimension, Z refers to the maximal iteration counts and $\lceil\cdot\rceil$ denotes the ceiling function. L shows the step size while n_1 represents the uniform distribution. A rabbit arbitrarily selects a burrow to hide in order to reduce the probability of getting hunted. The ith rabbit stays in the ith burrow which is mathematically expressed as follows.

$$b_{i,j}(z) = x_j(z) + H\cdot g\cdot x_j(z), i = 1, \cdots, n \text{ and } j = 1, \cdots, d \tag{7}$$

$$H = \frac{Z - z + 1}{Z}\cdot r_4 \tag{8}$$

$$n_2 \sim N(0,1) \tag{9}$$

$$g(k) = \begin{cases} 1 \ if \ k = j \\ 0 \ else \end{cases} \quad k = 1, \cdots, d. \tag{10}$$

In order to simulate these arbitrary hiding behaviors of the rabbits, the following mathematical equation is applied

$$v_i(z+1) = x_i(z) + R\cdot(r_4\cdot b_{i,r}(z) - x_i(z)), i = 1, \cdots, n \tag{11}$$

$$g_r(k) = \begin{cases} 1 \ if \ k = \lceil r_5\cdot d\rceil \\ 0 \ else \end{cases} \quad k = 1, \cdots d \tag{12}$$

$$b_{i,r}(z) = x_i(z) + H\cdot g_r\cdot x_i(z). \tag{13}$$

Here, $b_{i,r}$ denotes a randomly chosen burrow from the d set of burrows. r_4 and r_5 indicate two random integers within [0, 1]. Based on Equation (11), the ith rabbit arbitrarily chooses

a burrow from the d set of burrows to update the location. The location of the ith rabbit is updated, once the random hiding process and detour foraging strategy are completed.

$$x_i(z+1) = \begin{cases} x_i(z) & f(x_i(z)) \le f(v_i(z+1)) \\ v_i(\mathrm{z}+1) & f(x_i(z)) > f(v_i(z+1)). \end{cases} \tag{14}$$

An energy factor should be considered for stimulating the changeover from exploration to exploitation in the iteration, as given below.

$$A(z) = 4\Big(1 - \frac{z}{Z}\Big)\ln\frac{1}{r}. \tag{15}$$

The behavior of A over 1000 iterations. Once the energy feature $A(z) > 1$, then the detour foraging process takes place. Afterward, the rabbit tends to arbitrarily explore the area for forage during the exploration stage. When the energy aspect $A(z) \le 1$, then random hiding of the rabbit occurs. Afterward, the rabbit tends to arbitrarily exploit its burrows during the exploitation stage.

The ARO technique has a few benefits such as strong optimization performance, a simple process, and only a few parameters to be set. Further, it also contains points that can be enhanced including local optima avoidance, an increase in the convergence accuracy, and convergence speed. Since the population is comparatively dispersed, a large weighted value is allocated to speed up the global search capability of the model during the earlier phase of the iteration, using dynamic inertial weight. At the final stages, the module may vary the weight size values based on the distribution of individuals in the existing population. This is executed in such a manner that it can accelerate the convergence speed and finely search around the optimum solution. The current study presents an approach for improving the ARO techniques such as ARO with adaptive weight (IARO) based on the development of WOA.

Inertia weight is a crucial parameter in enhancing the ARO technique. The suitable weighted value can enhance the optimization capability of the model. Thus, the study presents a technique to change the weight value based on the population distribution as given below:

$$w = a_1 \cdot \left(P_{worst}^{j} - P_{best}^{j}\right) + \frac{a_2}{z} \cdot \left(x_i^{\max\, j} - x_i^{\min\, j}\right) \tag{16}$$

$$x_i^{msx\ j} = \max_{j\in[1,d]} x_i^j \quad i = 1, \cdots, n \tag{17}$$

$$x_i^{\max\, j} = \min_{j\in[1,d]} x_i^j i = 1, \cdots, n \tag{18}$$

$$a_1 = \cos(0.5\tau\tau \cdot r) \tag{19}$$

$$a_2 = 1 - a_1. \tag{20}$$

Here, t denotes the number of iterations of the existing population. P_{iworst} and P_{ibest} correspond to the location vectors of the worst and optimum rabbits in the existing population. The significant difference between the weighted and the presented inertia weighted is that this technique exploits the data of the existing population to update the upper and lower bounds so as to adaptively alter the searching space. Furthermore, this technique exploits two arbitrary coefficients to compromise the maximal individual distance and distance from all the dimensions. The adaptive change for the weight of the existing rabbit, from the arbitrary hidden upgrade location, is given herewith.

$$v_i(z+1) = w \cdot x_i(z) + R \cdot (r_4 \cdot b_{i,r}(z) - x_i(z)). \tag{21}$$

Based on the distribution of the existing population, the model can change the weight size, after presenting adaptive adjustment weighted strategies. At the beginning of the iteration, when the population falls as the local optimum, there exists a slight variance between the optimum and the worst solution. The value of $a_1 \cdot \left(P^j_{worst} - P^j_{best}\right)$ remains unaffected by population distribution. This term might develop a large weight value of w, in order to prevent getting trapped as a smaller searching range at the early stage of the iteration. The value of $a_1 \cdot \left(P^j_{worst} - P^j_{best}\right)$ slowly develops into a smaller value with an increased number of iterations and t. Further, its impact on the weight w also decreases. The design of $a_2 \cdot \left(x_i^{ma \times j} - x_i^{\min\, j}\right)/\mathrm{z}$ plays a major role in weight value w, if the algorithm does not achieve the optimum solution, it makes the model search with a large step size. The benefit of IARO can be defined in two parts.

Equation (13) is used to assess the quality of the received solution using the proposed IARO approach [27].

$$F_n = \alpha Error(P) + \beta \frac{|S|}{|A|}. \tag{22}$$

In Equation (22), P represents the group z of the inputs to the method. The importance of the features, selected from the population, is reflected by the value of $\in [0, 1]$, $\beta = 1 - \alpha$. The number of features selected and described as $|S|$ are lower than the overall number of features present in the data and are represented as $|A|$. The optimum method is the one that exploits the smallest feature to make an accurate classification.

3.2. Ensemble Learning Process

In order to predict the traffic flow, the ensemble learning process is involved and it contains GRU, BiGRU, and the LSTM models. The LSTM model is commonly utilized to overcome vanishing gradient problems [28]. Input, forget, and output gates are utilized for deciding the amount of data to be maintained from the new input and historical memory.

$$\hat{h} = \tanh\left(W^{\hat{h}x} x^{(t)} + W^{\hat{h}h} h^{(t-1)} + b_{\hat{h}}\right) \tag{23}$$

$$i^{(t)} = \sigma\left(W^{ix} x^{(t)} + W^{ih} h^{(t-1)} + b_i\right) \tag{24}$$

$$f^{(t)} = \sigma\left(W^{fx} x^{(t)} + W^{fh} h^{(t-1)} + b_f\right) \tag{25}$$

$$o^{(t)} = \sigma\left(W^{ox} x^{(t)} + W^{oh} h^{(t-1)} + b_o\right) \tag{26}$$

$$s^{(t)} = \hat{h} \odot i^{(t)} + s^{(t-1)} \odot f^{(t)} \tag{27}$$

$$h^{(t)} = \tanh\left(s^{(t)}\right) \odot o^{(t)}. \tag{28}$$

Here, t refers to time step t, i, f and 0 denotes the input, forget, and output gates correspondingly. $\odot$ indicates element-wise multiplication. b represents the bias and W denotes the weighted matrix. $x, \hat{h}, s$, and h show the input, short-term memory, long-term memory, and the output, correspondingly. The instinct of the term 'LSTM' is that the presented model exploits both short- and long-term memory vectors for encoding the in-sequence data. Further, it also exploits the gate processes to control the data flow. The outcome of the LSTM method is remarkable because it achieves better outcomes in NLP tasks as a backbone.

The GRU method is stimulated by the gating mechanism that is modeled by the following equations.

$$z^{(t)} = \sigma\left(W^z x^{(t)} + U^z h^{(t-1)} + b_z\right) \tag{29}$$

$$r^{(t)} = \sigma\left(W^r x^{(t)} + U^r h^{(t-1)} + b_r\right) \tag{30}$$

$$\hat{h} = \tanh\left(W^h x^{(t)} + U^h\left(r^{(t)} \odot h^{(t-1)}\right) + b_h\right) \tag{31}$$

$$h^{(t)} = \left(1 - z^{(t)}\right) \odot h^{(t-1)} + z^{(t)} \odot \hat{h}. \tag{32}$$

In these expressions, z and r represent the update and reset gates correspondingly. t denotes the time step, $\odot$ shows the element-wise multiplication. $x, \hat{h}$, and h indicate the input, candidate activation vector, and output, correspondingly. W and U are weight matrices while b shows the bias.

GRU and LSTM, the two variants of gating mechanisms, are similar to one another. However, unlike LSTM, the GRU mechanism does not individually gate the count of novel memory content being added. On the other hand, the GRU mechanism has lesser parameters that result in better generalization ability and fast convergence. Further, GRU has the ability to accomplish the best performance within a small dataset. BiGRU has two types of hidden layers: the former encoded data in the previous time step, whereas the latter encoder data in the flipped direction.

$$h^{(t)} = \sigma\left(W^{hx} x^{(t)} + W^{hh} h^{(t-1)} + b_h\right) \tag{33}$$

$$z^{(t)} = \sigma\left(W^{zx} x^{(t)} + W^{zz} z^{(t+1)} + b_z\right) \tag{34}$$

$$\hat{y}^{(t)} = softmax\left(W^{yh} h^{(t)} + W^{yz} z^{(t)} + b_y\right). \tag{35}$$

Here, h and z denote the hidden layers.

3.3. Hyperparameter Tuning

In the current research work, the GOA adjusts the hyperparameter values of the DL models. GOA is a bio-inspired optimization algorithm that simulates the swarming performance of grasshoppers in nature [29]. It drives by simulating the interaction and the movement of grasshoppers from a population to search for better performance. In this method, all the grasshoppers signify a great solution to the optimization issue. The positions of the grasshoppers, in the searching space, always get upgraded based on their fitness values and the stimulus of nearby grasshoppers. This combined movement encourages the exploration of the searching space and this method dynamically adjusts to determine the better regions with optimal fitness values.

The mathematical model of the grasshopper tends to procedure the swarms, with the location of the grasshopper in the swarm, demonstrating a possible solution. X_i, the location of the ith grasshopper is formulated by Equation (36) as a function of wind advection (A_i), social interaction (S_i), and gravity force (G_i).

$$X_i = S_i + G_i + A_i. \tag{36}$$

After ignoring the gravity element and the wind direction near the goal, the formula is changed to optimizer of the N grasshoppers and is formulated as given below.

$$X_i^d = c\left(\sum_{\substack{j=1 \\ j \neq i}}^{N} \frac{ub_d - lb_d}{2} s\left(\left|x_j^d - x_i^d\right|\right) \frac{x_j - x_i}{d_{ij}}\right) + \widehat{T}_d. \tag{37}$$

Here, f represents the intensity of attraction, d_{ij} denotes the distance between ith and jth grasshoppers, s_j indicates the strength of social forces, and l shows the attractive range scale.

$$d_{ij} = \left| d_j - d_i \right| \tag{38}$$

$$s(r) = fe^{\frac{-r}{l}} - e^{-r}. \tag{39}$$

The modification of l and f makes it feasible to model the social behaviors of the grasshopper by modifying a set of features such as the repulsion, comfort, and attraction zones of the swarm. The s function returns the value closer to 0 for a distance more than 10 and the distance is mapped towards the interval. ub_d and ib_d indicate the upper and lower boundaries at the dth dimension $s(r) = fe^{\frac{-r}{l}} - e^{-r}$. $\widehat{T}_d$ represents the value of the dth dimension from the goal. The c parameter is the minimizing coefficient to shrink the attraction, comfort, and repulsion zones. The c parameter minimizes the comfort zone with all the iterations, where $c$max and $c$min represent the maximal and minimal values correspondingly, l shows the current value and L denotes the maximal number of iterations:

$$c = cmax - l\frac{c\ \max - c\ \min}{L}. \tag{40}$$

Therefore, the grasshopper is initialized at random locations. At last, the grasshopper reaches the stop-moving and comfort zones. The next position of the grasshoppers is a function of its existing position, target location, and the location of other grasshoppers from the swarm.

In this case, the GOA is utilized to find the hyperparameters present in the MABLSTM approach. The *MSE* is assumed as the main function and is determined as follows.

$$MSE = \frac{1}{T}\sum_{j=1}^{L}\sum_{i=1}^{M}\left(y_j^i - d_j^i\right)^2. \tag{41}$$

Here, M and L signify the outcome values of the layer and data correspondingly, y_j^i and d_j^i imply the accomplished and suitable values to the jth unit in the outcome layer of the network at time t, correspondingly.

4. Results and Discussion

In this section, the TFP results of the IAROEL-TFMS model are discussed in detail. The proposed model was validated using the traffic data containing all 30 s raw sensor data for 30 days. The traffic data, collected during the first 10 days, was used as a training dataset the dataset containing the data for the remaining 20 days was utilized as a testing dataset. In this experiment, the data groups consist of 15 min of aggregated data in vehicles per 15 min (veh per 15 min). Thereby, 96 data groups are available for each day. Before the calculation, the data groups were normalized and the data was rendered in the range of 0 to 1.

Table 1 and Figure 2 highlight the comparative analysis results achieved by the proposed IAROEL-TFMS approach in terms of MAPE. The simulation results highlight the effectual prediction performance of the IAROEL-TFMS model. It can also be stated that the proposed IAROEL-TFMS model achieved low MAPE values under several iterations.

Table 2 and Figure 3 exhibit the TFP outcomes of the IAROEL-TFMS approach over varying time indices. The experimental outcomes infer the maximum performance of the IAROEL-TFMS approach under all the time indices. For instance, with a 10-time index and an actual value of 136, the proposed IAROEL-TFMS model obtained the prediction values such as 129, 129, 117, 87, and 81 underruns 1–5, respectively. Simultaneously, with a 20-time index and an actual value of 882, the IAROEL-TFMS approach attained the prediction values such as 892, 859, 861, 923, and 917 underruns 1–5, respectively. At the same time, with a 70-time index and an actual value of 853, the IAROEL-TFMS system attained the prediction values such as 865, 826, 826, 879, and 886 underruns 1–5, correspondingly. Finally,

with a 100-time index and an actual value of 136, the IAROEL-TFMS algorithm obtained the prediction values such as 609, 629, 594, 604, and 557 underruns 1–5, correspondingly.

In Table 3, the comprehensive prediction analysis results achieved by the proposed IAROEL-TFMS method and the existing approaches are listed [30,31]. Figure 4 shows the RMSE results of the IAROEL-TFMS method and the existing approaches. The figure infers the superior results of the IAROEL-TFMS model with minimal RMSE values. On Lag 1, the IAROEL-TFMS model accomplished a low RMSE of 20.5057, whereas the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN techniques achieved high RMSE values such as 42.6758, 39.5084, 38.0203, 32.7534, and 26.9257, correspondingly. Moreover, on Lag 3, the IAROEL-TFMS system obtained the least RMSE of 16.4539, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN approaches accomplished the highest RMSE values such as 51.706, 47.4003, 45.2506, 22.8315, and 16.5139, respectively. Finally, on Lag 5, the IAROEL-TFMS algorithm realized the least RMSE of 17.8412 while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN approaches producing high RMSE values such as 58.9394, 55.5521, 53.4518, 28.7079, and 23.4312, correspondingly.

Table 1. MAPE outcomes of the IAROEL-TFMS algorithm under various iterations.

MAPE (%)			
No. of Iterations	**Best Fitness**	**Average Fitness**	**Worst Fitness**
0	1.36	4.84	7.34
10	0.56	2.57	6.88
20	0.18	0.71	4.41
30	0.36	0.77	4.08
40	0.40	1.16	4.39
50	0.22	0.93	3.76
60	0.30	0.42	3.27
70	0.20	1.06	3.05
80	0.28	0.85	2.55
90	0.13	0.16	2.66
100	0.11	0.80	2.48

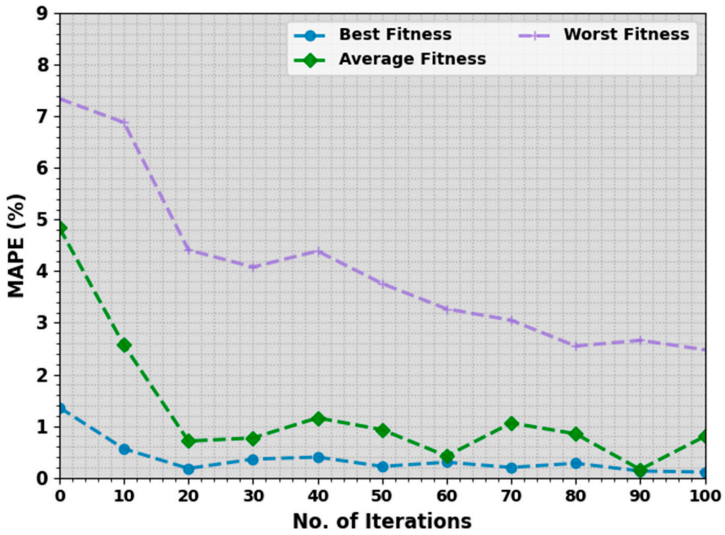

Figure 2. MAPE analysis outcomes of the IAROEL-TFMS approach under various iterations.

Table 2. TFP outcomes of the IAROEL-TFMS approach under varying time indices.

Traffic Flow						
Time Index	**Actual**	**Predicted**				
		Run1	**Run2**	**Run3**	**Run4**	**Run5**
0	188	227	232	242	235	258
10	136	129	129	117	87	81
20	882	892	859	861	923	917
30	958	965	977	977	922	936
40	722	697	657	649	653	619
50	765	801	828	797	799	783
60	955	972	989	1000	987	949
70	853	865	826	826	879	886
80	967	1015	1034	1025	1006	1045
90	1036	1033	1039	1012	960	951
100	609	629	594	604	557	545

Table 3. Comparative outcomes of the IAROEL-TFMS system and other methods.

No. of Lags	GA-LSSVM	PSO-LSSVM	FFO-LSSVM	Hybrid-LSSVM	AST2FP-OHDBN	IAROEL-TFMS
Root-Mean-Square Error						
Lag = 1	42.6758	39.5084	38.0203	32.7534	26.9257	20.5057
Lag = 2	46.9715	42.8572	41.0955	29.5013	23.6656	17.8856
Lag = 3	51.706	47.4003	45.2506	22.8315	16.5139	16.4539
Lag = 4	55.3336	51.0474	49.1178	27.9114	22.7806	22.6906
Lag = 5	58.9394	55.5521	53.4518	28.7079	23.4312	17.8412
Mean Absolute Error						
Lag = 1	32.3363	24.7114	22.3607	21.8427	15.477	14.477
Lag = 2	39.6895	32.0517	25.241	19.2498	12.3658	11.7658
Lag = 3	45.2973	39.4341	31.6977	23.5686	17.8232	16.1232
Lag = 4	51.9591	47.1757	37.1099	27.4439	20.5398	21.6398
Lag = 5	57.1088	52.4225	44.582	24.4347	19.7508	18.1508
Equal Coefficient						
Lag = 1	95.57	95.69	95.79	97.92	98.35	99.09
Lag = 2	95.37	95.55	95.74	97.88	98.49	98.95
Lag = 3	95.13	95.31	95.43	98.54	98.77	99.05
Lag = 4	94.93	95.13	95.26	98.39	98.63	99.15
Lag = 5	94.77	94.89	95.07	97.24	98.61	99.29

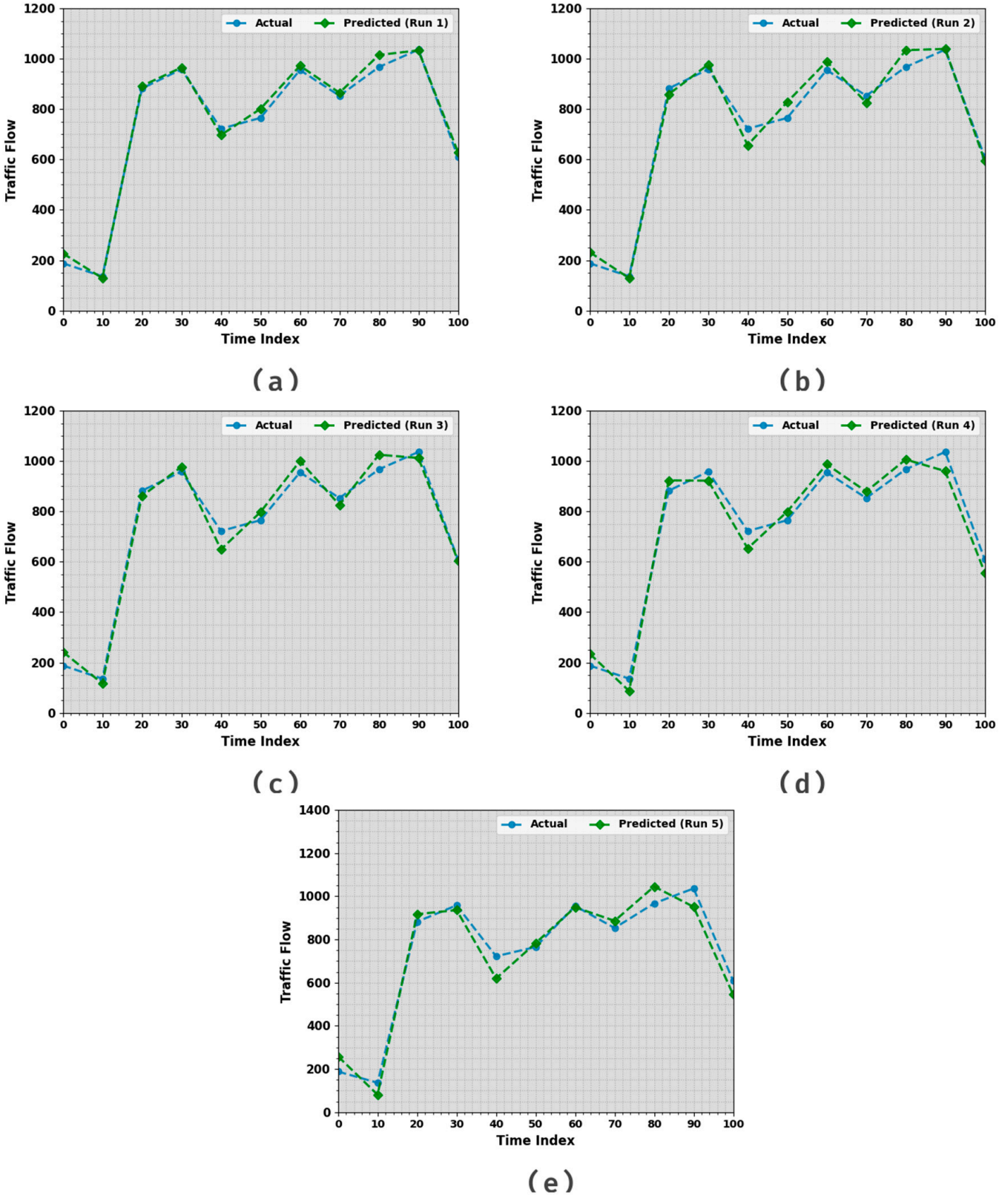

Figure 3. TFP outcomes of the IAROEL-TFMS system: (**a**) Run1, (**b**) Run2, (**c**) Run3, (**d**) Run4, and (**e**) Run5.

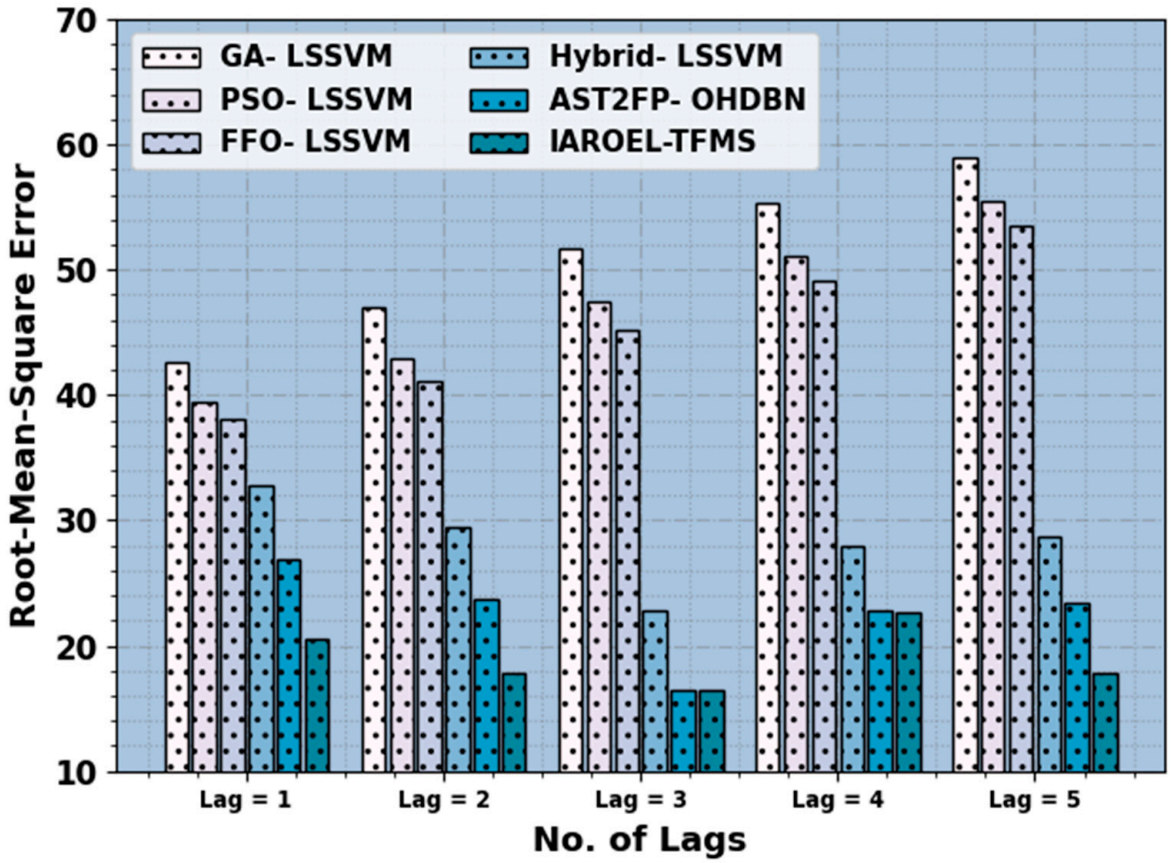

Figure 4. RMSE outcomes of the IAROEL-TFMS system and other methods.

Figure 5 shows the MAE outcomes of the IAROEL-TFMS and other existing approaches. The outcome implies the superior performance of the IAROEL-TFMS algorithm with minimal MAE values. On Lag 1, the IAROEL-TFMS methodology accomplished the least MAE of 14.477, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN techniques produced the maximum MAE values such as 32.3363, 24.7114, 22.3607, 21.8427, and 15.477, correspondingly. Furthermore, on Lag 3, the IAROEL-TFMS model accomplished a minimal MAE of 16.1232, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN systems produced high MAE values such as 45.2973, 39.4341, 31.6977, 23.5686, and 17.8232, respectively. Lastly, on Lag 5, the IAROEL-TFMS method achieved a low MAE of 18.1508, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN methods achieved superior MAE values such as 57.1088, 52.4225, 44.582, 24.4347, and 19.7508, correspondingly.

Figure 6 showcases the EC results achieved by the proposed IAROEL-TFMS and other existing algorithms. The result portrays the superior outcomes of the IAROEL-TFMS methodology with higher EC values. On Lag 1, the IAROEL-TFMS approach reached a higher EC of 99.09, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN methods produced minimal EC values such as 95.57, 95.69, 95.79, 97.92, and 98.35, correspondingly. Moreover, on Lag 3, the IAROEL-TFMS approach accomplished a superior EC of 99.05, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN systems produced low EC values such as 95.13, 95.31, 95.43, 98.54, and 98.77, correspondingly. At last, on Lag 5, the IAROEL-TFMS model accomplished an enhanced EC of 99.29, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN methods yielded low EC values such as 94.77, 94.89, 95.07, 97.24, and 98.61, respectively.

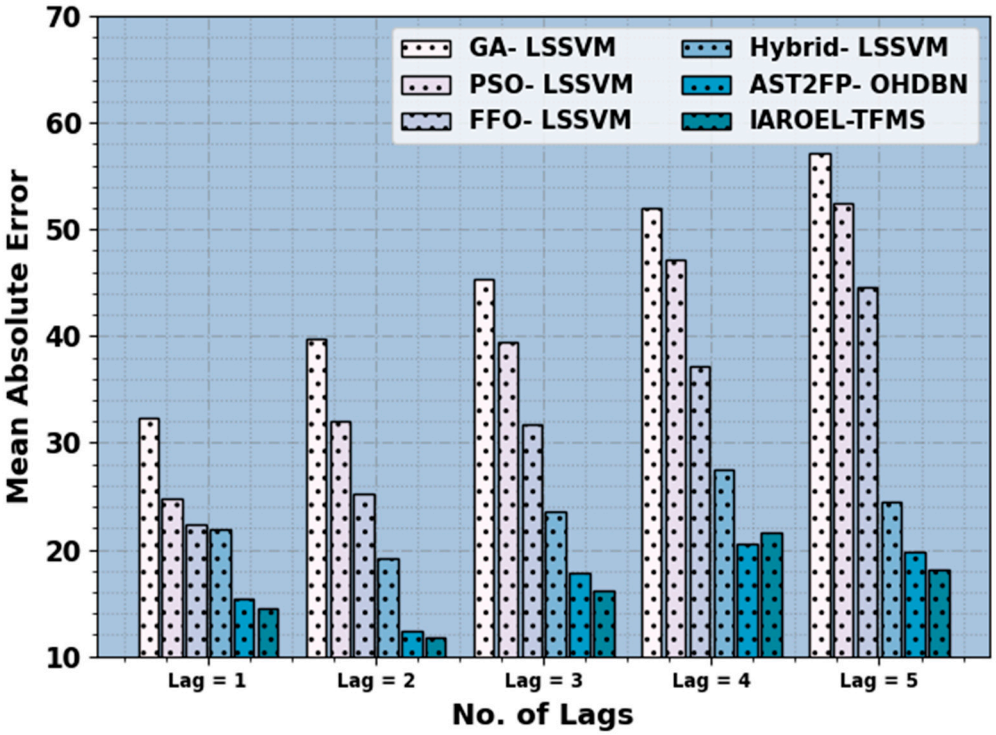

Figure 5. MAE outcomes of the IAROEL-TFMS system with other methods.

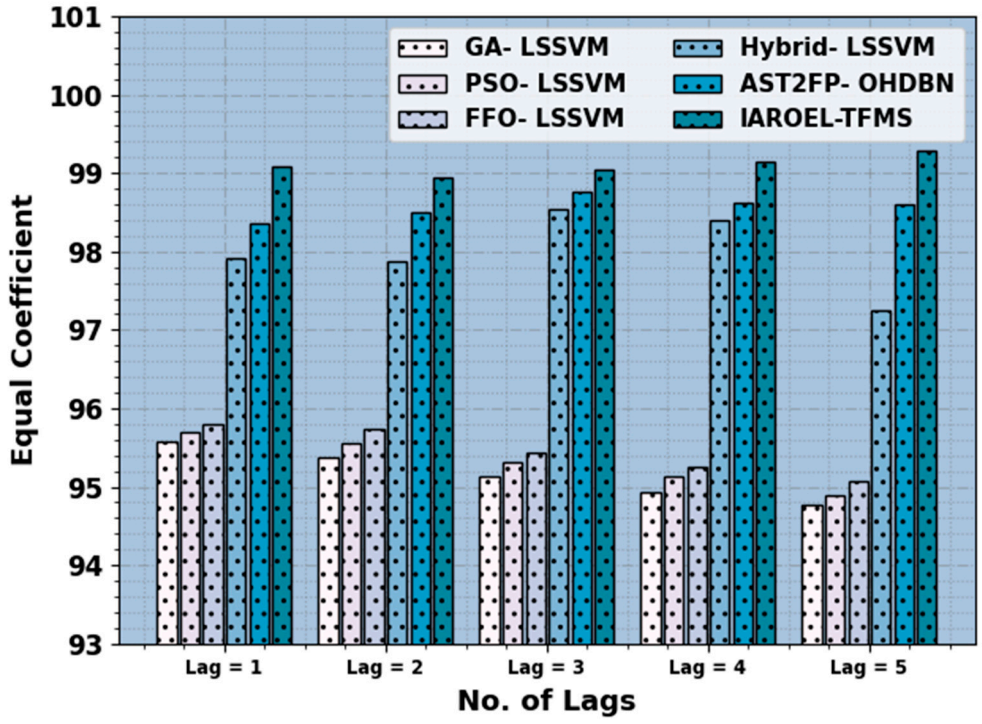

Figure 6. EC outcomes of the IAROEL-TFMS system with other methods.

Table 4 and Figure 7 demonstrate the RT analytical outcomes of the IAROEL-TFMS and other existing approaches. The figure infers the superior results achieved by the proposed IAROEL-TFMS model with minimal RT values. On Lag 1, the IAROEL-TFMS approach achieved a minimal RT of 2.09 s, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN methods achieved high RT values such as 19.53 s, 11.29 s, 14.25 s, 9.8 s, and 3.84 s, correspondingly. Moreover, on Lag 3, the IAROEL-TFMS

system attained a low RT of 4.03 s, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN systems accomplished the maximum RT values such as 20.33 s, 11.67 s, 15.06 s, 10.02 s, and 5.13 s, respectively. Finally, on Lag 5, the IAROEL-TFMS algorithm accomplished a low RT of 4.26 s, while the existing Ga+SVM, PSO+LSSVM, FFO+LSSVM, HYBRID+LSSVM, and AST2FP+OHDBN algorithms yielded high RT values such as 22 s, 12.24 s, 15.31 s, 10.44 s, and 2.07 s respectively.

Table 4. RT outcomes of the IAROEL-TFMS system with other methods.

	Running Time (s)					
No. of Lags	**GA-LSSVM**	**PSO-LSSVM**	**FFO-LSSVM**	**Hybrid-LSSVM**	**AST2FP-OHDBN**	**IAROEL-TFMS**
Lag = 1	19.53	11.29	14.25	9.8	3.84	2.09
Lag = 2	20.01	11.64	14.45	9.98	3.56	2.58
Lag = 3	20.33	11.67	15.06	10.02	5.13	4.03
Lag = 4	20.52	12.18	15.28	10.03	5.05	3.69
Lag = 5	22	12.24	15.31	10.44	5.07	4.26

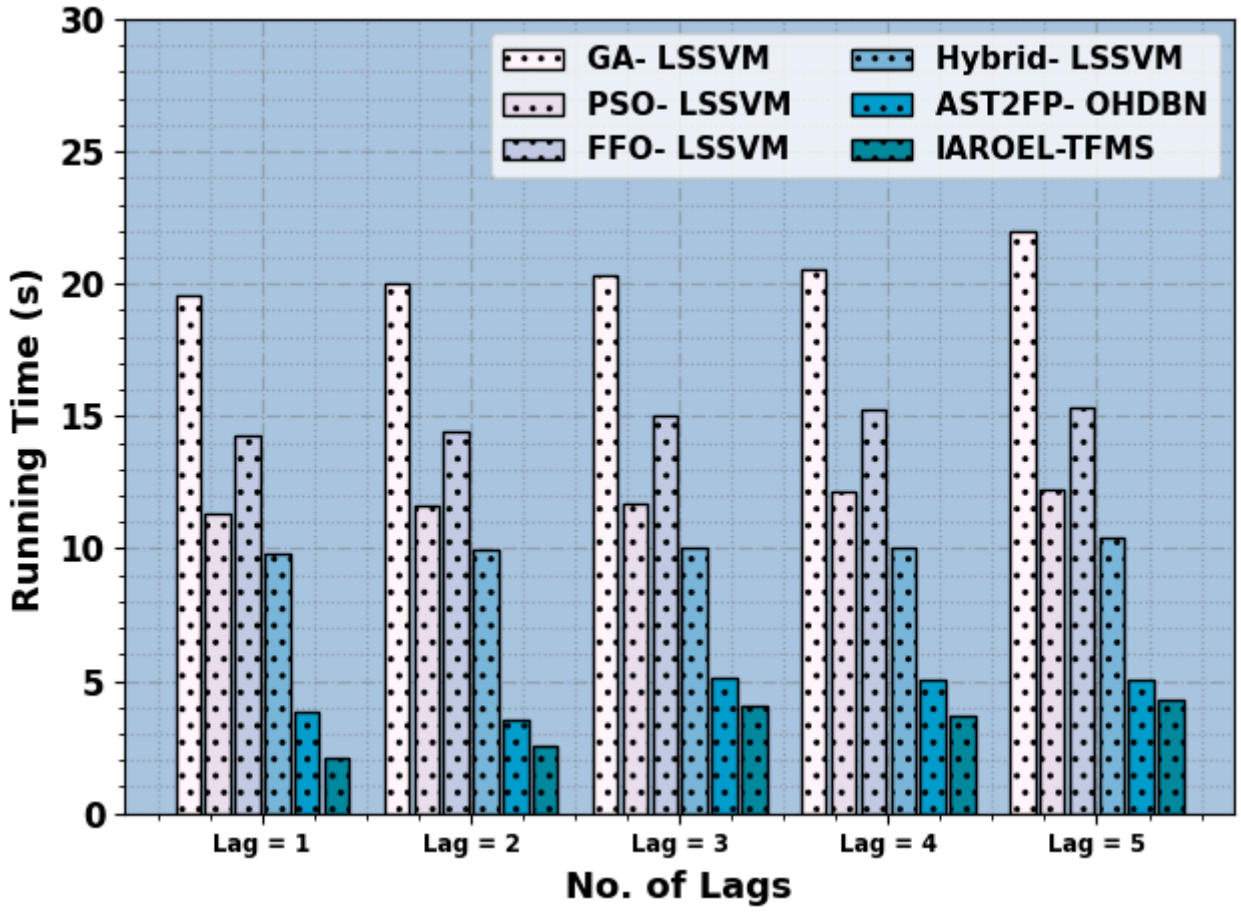

Figure 7. RT outcome of IAROEL-TFMS system with other methods.

5. Conclusions

In the current study, the authors have designed a novel IAROEL-TFMS approach for traffic flow monitoring. The major intention of the proposed IAROEL-TFMS technique is to employ the feature subset selection process with an optimal ensemble learning process for forecasting the traffic flow. To accomplish this, the IAROEL-TFMS technique comprises

three stages of operations, namely, IARO-based feature subset selection, ensemble classification, and GOA-based hyperparameter tuning. The combination of the IARO system with ensemble learning represents a significant advancement in traffic flow monitoring, paving the way for a highly efficient and intelligent transportation network. In order to highlight the improved prediction results of the IAROEL-TFMS algorithm, an extensive range of simulations was conducted. The simulation outcomes reported the supremacy of the IAROEL-TFMS approach over other existing approaches. Therefore, the IAROEL-TFMS technique can be utilized for effective traffic flow monitoring. In the future, the IAROEL-TFMS system can be extended by including outlier removal approaches.

Author Contributions: Conceptualization, M.R., A.G.F. and A.A.-M.A.-G.; Methodology, M.R., A.G.F. and A.A.-M.A.-G.; Software, L.M. and D.A.; Validation, H.A.A., D.A. and A.A.-M.A.-G.; Investigation, A.G.F. and A.A.-M.A.-G.; Resources, H.A.A., L.M. and D.A.; Data curation, H.A.A., L.M., D.A. and A.G.F.; Writing—original draft, M.R.; Writing—review & editing, H.A.A., L.M. and D.A.; Visualization, H.A.A., L.M. and D.A.; Supervision, A.G.F. and A.A.-M.A.-G.; Project administration, M.R.; Funding acquisition, M.R. All authors have read and agreed to the published version of the manuscript.

Funding: The Deanship of Scientific Research (DSR) at King Abdulaziz University (KAU), Jeddah, Saudi Arabia, funded this project under grant no. (PH: 014-351-1443).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The authors confirm that the data supporting the findings of this study are available at https://doi.org/10.3390/app122110828.

Acknowledgments: The authors gratefully acknowledge technical and financial support provided by the Deanship of Scientific Research (DSR), King Abdulaziz University (KAU), Jeddah, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dai, F.; Huang, P.; Mo, Q.; Xu, X.; Bilal, M.; Song, H. ST-InNet: Deep Spatio-Temporal Inception Networks for Traffic Flow Prediction in Smart Cities. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 19782–19794. [CrossRef]
2. Huo, G.; Zhang, Y.; Wang, B.; Gao, J.; Hu, Y.; Yin, B. Hierarchical Spatio–Temporal Graph Convolutional Networks and Transformer Network for Traffic Flow Forecasting. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 3855–3867. [CrossRef]
3. Shu, W.; Cai, K.; Xiong, N.N. A short-term traffic flow prediction model based on an improved gate recurrent unit neural network. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 16654–16665. [CrossRef]
4. Shukla, A.; Bhattacharya, P.; Tanwar, S.; Kumar, N.; Guizani, M. Dwara: A deep learning-based dynamic toll pricing scheme for intelligent transportation systems. *IEEE Trans. Veh. Technol.* **2020**, *69*, 12510–12520. [CrossRef]
5. Manimurugan, S.; Almutairi, S. Non-divergent traffic management scheme using classification learning for smart transportation systems. *Comput. Electr. Eng.* **2023**, *106*, 108581.
6. Chen, C.; Liu, B.; Wan, S.; Qiao, P.; Pei, Q. An edge traffic flow detection scheme based on deep learning in an intelligent transportation system. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 1840–1852. [CrossRef]
7. Fang, W.; Zhuo, W.; Song, Y.; Yan, J.; Zhou, T.; Qin, J. Δfree-LSTM: An error distribution free deep learning for short-term traffic flow forecasting. *Neurocomputing* **2023**, *526*, 180–190. [CrossRef]
8. Chen, M.Y.; Chiang, H.S.; Yang, K.J. Constructing cooperative intelligent transport systems for travel time prediction with deep learning approaches. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 16590–16599. [CrossRef]
9. Haghighat, A.K.; Ravichandra-Mouli, V.; Chakraborty, P.; Esfandiari, Y.; Arabi, S.; Sharma, A. Applications of deep learning in intelligent transportation systems. *J. Big Data Anal. Transp.* **2020**, *2*, 115–145. [CrossRef]
10. Goswami, S.; Kumar, A. Traffic Flow Prediction Using Deep Learning Techniques. In *Computing Science, Communication and Security: Third International Conference, COMS2 2022, Gujarat, India, 6–7 February 2022*; Revised Selected Papers; Springer International Publishing: Cham, Switzerland, 2022; pp. 198–213.
11. Djenouri, Y.; Belhadi, A.; Srivastava, G.; Lin, J.C.W. Hybrid graph convolution neural network and branch-and-bound optimization for traffic flow forecasting. *Future Gener. Comput. Syst.* **2023**, *139*, 100–108. [CrossRef]
12. Han, L.; Huang, Y.S. Short-term traffic flow prediction of road network based on deep learning. *IET Intell. Transp. Syst.* **2020**, *14*, 495–503. [CrossRef]
13. Alkinani, M.H.; Almazroi, A.A.; Adhikari, M.; Menon, V.G. Design and analysis of logistic agent-based swarm-neural network for intelligent transportation system. *Alex. Eng. J.* **2022**, *61*, 8325–8334. [CrossRef]

14. Qi, T.; Chen, L.; Li, G.; Li, Y.; Wang, C. FedAGCN: A traffic flow prediction framework based on federated learning and Asynchronous Graph Convolutional Network. *Appl. Soft Comput.* **2023**, *138*, 110175. [CrossRef]
15. Cheng, Z.; Lu, J.; Zhou, H.; Zhang, Y.; Zhang, L. Short-term traffic flow prediction: An integrated method of econometrics and hybrid deep learning. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 5231–5244. [CrossRef]
16. Chan, R.K.C.; Lim, J.M.Y.; Parthiban, R. A neural network approach for traffic prediction and routing with missing data imputation for intelligent transportation system. *Expert Syst. Appl.* **2021**, *171*, 114573. [CrossRef]
17. Wu, Y.; Wu, L.; Cai, H. A deep learning approach to secure vehicle to road side unit communications in intelligent transportation system. *Comput. Electr. Eng.* **2023**, *105*, 108542. [CrossRef]
18. Ma, C.; Zhao, Y.; Dai, G.; Xu, X.; Wong, S.C. A novel STFSA-CNN-GRU hybrid model for short-term traffic speed prediction. *IEEE Trans. Intell. Transp. Syst.* **2022**, *24*, 3728–3737. [CrossRef]
19. Neelakandan, S.; Prakash, M.; Bhargava, S.; Mohan, K.; Robert, N.R.; Upadhye, S. Optimal stacked sparse autoencoder based traffic flow prediction in intelligent transportation systems. In *Virtual and Augmented Reality for Automobile Industry: Innovation Vision and Applications*; Springer International Publishing: Cham, Switzerland, 2022; pp. 111–127.
20. Liang, M.; Liu, R.W.; Zhan, Y.; Li, H.; Zhu, F.; Wang, F.Y. Fine-grained vessel traffic flow prediction with a spatio-temporal multigraph convolutional network. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 23694–23707. [CrossRef]
21. Xia, M.; Jin, D.; Chen, J. Short-term traffic flow prediction based on graph convolutional networks and federated learning. *IEEE Trans. Intell. Transp. Syst.* **2022**, *24*, 1191–1203. [CrossRef]
22. Zhang, W.; Yao, R.; Du, X.; Liu, Y.; Wang, R.; Wang, L. Traffic flow prediction under multiple adverse weather based on self-attention mechanism and deep learning models. *Phys. A Stat. Mech. Its Appl.* **2023**, *625*, 128988. [CrossRef]
23. Liu, C.; Ke, L. Cloud assisted Internet of things intelligent transportation system and the traffic control system in the smart city. *J. Control. Decis.* **2023**, *10*, 174–187. [CrossRef]
24. Kong, J.; Fan, X.; Jin, X.; Lin, S.; Zuo, M. A variational bayesian inference-based En-Decoder framework for traffic flow prediction. *IEEE Trans. Intell. Transp. Syst.* **2023**. *early access*. [CrossRef]
25. Wen, Y.; Xu, P.; Li, Z.; Xu, W.; Wang, X. RPConvformer: A novel Transformer-based deep neural networks for traffic flow prediction. *Expert Syst. Appl.* **2023**, *218*, 119587. [CrossRef]
26. Cao, Q.; Wang, L.; Zhao, W.; Yuan, Z.; Liu, A.; Gao, Y.; Ye, R. Vibration State Identification of Hydraulic Units Based on Improved Artificial Rabbits Optimization Algorithm. *Biomimetics* **2023**, *8*, 243. [CrossRef]
27. Alhussan, A.A.; Abdelhamid, A.A.; Towfek, S.K.; Ibrahim, A.; Eid, M.M.; Khafaga, D.S.; Saraya, M.S. Classification of Diabetes Using Feature Selection and Hybrid Al-Biruni Earth Radius and Dipper Throated Optimization. *Diagnostics* **2023**, *13*, 2038. [CrossRef]
28. Ni, J.; Young, T.; Pandelea, V.; Xue, F.; Cambria, E. Recent advances in deep learning based dialogue systems: A systematic survey. *Artif. Intell. Rev.* **2023**, *56*, 3055–3155. [CrossRef]
29. Simumba, N.; Okami, S.; Kodaka, A.; Kohtake, N. Multiple objective metaheuristics for feature selection based on stakeholder requirements in credit scoring. *Decis. Support Syst.* **2022**, *155*, 113714. [CrossRef]
30. Mohammed, G.P.; Alasmari, N.; Alsolai, H.; Alotaibi, S.S.; Alotaibi, N.; Mohsen, H. Autonomous Short-Term Traffic Flow Prediction Using Pelican Optimization with Hybrid Deep Belief Network in Smart Cities. *Appl. Sci.* **2022**, *12*, 10828. [CrossRef]
31. Luo, C.; Huang, C.; Cao, J.; Lu, J.; Huang, W.; Guo, J.; Wei, Y. Short-term traffic flow prediction based on least square support vector machine with hybrid optimization algorithm. *Neural Process. Lett.* **2019**, *50*, 2305–2322. [CrossRef]