Journal of Computer Science and Artificial Intelligence
ISSN: 3078-2309 (Online) | Vol. 1, No. 1, 2024

Vehicle Travel Time Prediction Based on SSA-LSTM

Yuanrong Wang, Tingmei Wang *, Guoxia Dong

Beijing Union University, Beijing, 100101, China

* **Corresponding author**: Tingmei Wang (Email: yykjttingmei@buu.edu.cn)

Abstract: Highway travel time, defined as the time difference between a vehicle entering and exiting the highway, is primarily influenced by factors such as the highway's characteristics, current road conditions, and the driver's behavior. Travel time is a crucial parameter for assessing the operational state of a highway and serves as an important reference for both drivers when selecting routes and for traffic managers in implementing control measures. Therefore, predicting highway travel time is of significant importance. Long Short-Term Memory (LSTM) is a type of recurrent neural network, an improved version of the traditional recurrent neural network, known for its excellent performance in predicting sequential data, such as time-series data on highway travel times. In operations research, search optimization algorithms have been a key area of focus in conjunction with machine learning. The Sparrow Search Algorithm (SSA), a relatively new search optimization algorithm, boasts high convergence speed and accuracy, along with strong global search capabilities, reducing the risk of getting trapped in local optima. This study utilizes vehicle entry and exit data collected by the ETC system at the Jin-Gang Expressway toll stations to analyze the highway's basic information. Building upon the traditional LSTM model, we integrate the SSA to optimize the LSTM model parameters. By using SSA to search for the optimal parameters, we form a new SSA-LSTM model. Finally, we apply this model to predict travel time data, showing a 120% improvement in prediction accuracy compared to the original LSTM model.

Keywords: Highwa, LSTM Neural Network, Search Optimization Algorithm, SSA Model.

1. Introduction

In recent years, with the rapid development of the global economy and the significant improvement in people's living standards, the number of motor vehicles has shown a continuous growth trend. By the end of 2023, the total number of motor vehicles in China had reached an astonishing 435 million. At the same time, the total length of highways in China has approached 177,000 kilometers, ranking first in the world. However, even so, on highways, especially during times when driving is ideal, such as in the mornings or evenings, traffic often experiences an imbalance between supply and demand due to the spatial and temporal clustering of vehicles. This leads to road congestion and driving difficulties. To address this phenomenon, there are two main approaches: one is to alleviate the imbalance between traffic supply and demand by restricting vehicle flow or increasing highway capacity; the other is through traffic management measures to guide drivers to avoid peak hours or congested sections, thereby rationally distributing traffic flow and improving the overall efficiency of the transportation system. The second approach requires a certain understanding of traffic information, which is where highway travel time plays a crucial role. As a key piece of traffic data, predicting highway travel time, especially short-term travel time, becomes particularly important.

Regarding highway travel time prediction, research in European and American countries has been relatively advanced due to their earlier economic development and rapid highway construction. For example, as early as 1993, Nelson and Palacharla attempted to address travel time prediction using neural network methods [1]. However, due to limitations in computing power and algorithms at that time, the results were not ideal, and the focus remained on using operations research models for travel time prediction. With developments over recent decades, the methods for predicting highway travel time have diversified, including a range of operations research and machine learning approaches. For instance, Ladino defined dynamic and instantaneous travel time concepts and found that the difference between the two increased significantly when future traffic flow conditions changed. They utilized the K-means clustering algorithm on historical data for dynamic clustering and used the occurrence time of historical data to predict current travel times, eventually combining the two types of travel time predictions using Kalman filtering [2]. Teresa considered GPS information and vehicle travel logs, using Euclidean and Manhattan distances, and K-means clustering methods to analyze travel time and predict vehicle travel times based on data similarity [3]. Xianyuan Zhan and Satish estimated travel time on roads using large-scale vehicle travel time databases, applying Bayesian fusion models for feasibility and effectiveness studies [4]. Anton Bezuglov, Gurcan Comert, and Anton Bezuglov applied grey system models to predict short-term highway travel times and tested them using data from California, USA, finding that grey models were easier to use and more adaptable to sudden external disturbances compared to nonlinear time series models [5]. Duan et al. improved the LSTM model and trained it with travel time data from 66 sections of highways in the UK, validating the model's predictive performance and obtaining relatively favorable results [6]. Li Meng et al. proposed a combined prediction algorithm based on KNN and RBF neural networks, using semi-distance methods to convert detector data into travel time and considering factors such as historical travel time, current traffic conditions, and weather data to predict travel time, focusing on travel time prediction under rainy conditions [7].

Other prediction methods have also been diverse. Zhang Juan et al. studied travel time prediction on weekdays and under good weather conditions using license plate recognition technology combined with the support vector machine model. The average speed of four time periods (5 minutes each) prior to the prediction time was used as input, and the model

learned from historical data to predict the average vehicle speed for the target time period [8]. Li Songjiang et al. classified travel behavior using variables such as travel date, time period, vehicle type, weather, and historical travel time, and predicted travel times based on entry and exit data from toll stations using BP neural networks [9]. Zhang Weiwei and others considered spatial correlation on highways and used LSTM neural networks for travel time prediction, comparing it with BP neural networks and highlighting the advantages of LSTM [10]. Ghulam et al. combined GRU and LSTM models to predict urban road travel time, improving prediction accuracy [11]. Li et al. integrated quantile regression models with LSTM to estimate travel time on road segments [12]. Jia et al. combined the CEEMDAN algorithm with LSTM, improving the prediction accuracy of highway travel times [13].

In summary, both domestic and international scholars have explored numerous methods for travel time prediction. These methods can generally be categorized into two types: model-driven and data-driven approaches. Model-driven methods involve using traffic flow characteristics to model and describe overall traffic, such as queuing theory and cellular automata models. Data-driven methods, on the other hand, make predictions based on large amounts of data in the absence of a well-defined model, such as neural networks and support vector regression. Recently, many scholars have tended to combine both approaches to enhance the accuracy and reliability of predictive models.

2. Problem Description

The data used in this study is sourced from the electronic toll collection (ETC) system and the manual semi-automatic toll collection lanes (MTC) at the southeastern end of the Jin-Gang Expressway, which recorded vehicle entry and exit data in August 2018. After basic data processing, the average travel time for vehicles on the road every half hour was calculated, forming a continuous time-series dataset.

Based on this data, modifications were made to the LSTM model (single input, single output) to generate parameter interfaces, which were then integrated into the Sparrow Search Algorithm (SSA). Using a metaheuristic search optimization approach, the optimal parameters for the overall model were identified, leading to a more accurate single-input, single-output SSA-LSTM travel time prediction model. This model can then be used to achieve more precise travel time predictions. The specific process is shown in Figure 1.

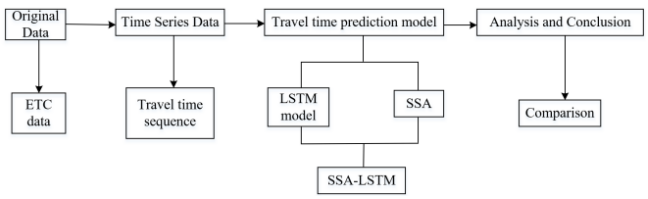

Figure 1. Structure of a Single Neuron in an LSTM Neural Network

3. Model Construction

3.1. Long Short-Term Memory Neural Network

The LSTM prediction model is a specialized type of RNN used for time series analysis. It introduces a gating structure that determines whether the information is relevant, controlling the speed at which information is accumulated. This helps to mitigate the issues of gradient explosion and vanishing gradients. The primary gating structures include the input gate, forget gate, and output gate. At its core, the model features a Sigmoid neural network layer and a pointwise multiplication operation. The output of the Sigmoid layer ranges from 0 to 1, where 0 means, no information is allowed to pass, and 1 means any amount of information can pass. This structure allows for the memory and updating of new information, addressing the issue of long-term dependencies. The structure of a single LSTM neuron is shown in Figure 2. Each LSTM unit consists of a cell state, which includes both long-term and short-term states, as well as the input gate, forget gate, and output gate.

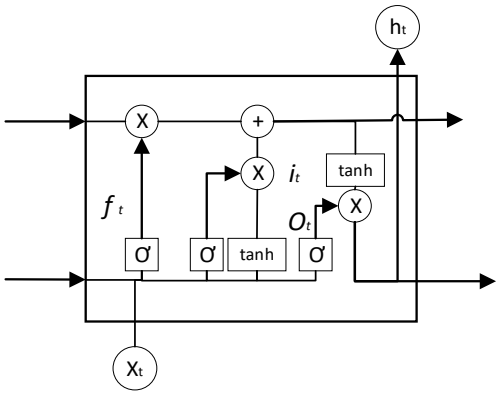

Figure 2. Structure of a Single Neuron in an LSTM Neural Network

Where

$$h_t = O_t \cdot \tanh(c_t) \quad (1)$$

$$O_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (2)$$

$$c_t = f_t c_{t-1} + i_t \tilde{c}_t \quad (3)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (4)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (5)$$

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1}) \quad (6)$$

In the above equation, f_t, i_t and O_t represent the forget gate, input gate, and output gate at time t, respectively; $\tilde{c}_t$ and c_t denote the contents stored in the memory cell at time t, x_t is the input vector at time t; h_t is the output of the hidden layer; σ represents the Sigmoid function; and W, U, and b are the weights involved in the computations.

The training process of the LSTM model is shown in Figure 3.

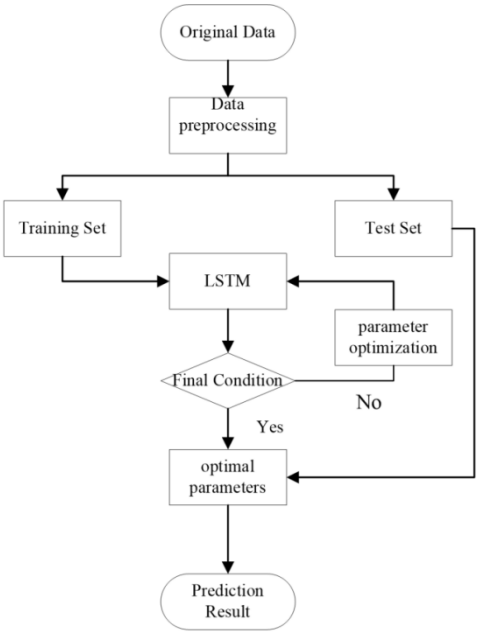

Figure 3. LSTM forecast flow chart

(1) Preprocess the training data (e.g., splitting the dataset into training and testing sets, data normalization, etc.).

(2) Initialize the network's weight, bias parameters, and hyperparameters.

(3) Run the LSTM, performing the forward training process to obtain the LSTM-based predicted values.

(4) Calculate the error between the predicted values and the actual values.

(5) Check if the error threshold or maximum number of iterations has been reached. If so, end the training; otherwise, update the parameters and return to step 3 for further iteration.

(6) Finally, present the prediction results.

3.2. Sparrow Search Algorithm

The Sparrow Search Algorithm (SSA) was proposed in 2020 by Xue and Shen from Donghua University. It simulates the foraging and anti-predation behaviors of sparrow flocks to search for the optimal solution of a function or model.

In the foraging process, sparrow individuals are typically divided into two types: explorers and followers. In their natural state, individuals monitor each other, and followers often compete for food resources from high-consumption peers to improve their own foraging success. At the same time, all individuals remain alert to potential predators in their environment.

The overall model is governed by the following six rules:

In the population, explorers generally have higher energy reserves and are responsible for searching areas rich in food resources, providing directions and areas for the followers to forage. In the algorithm, the energy reserve is directly related to the sparrow's fitness value.

(1) Once a sparrow detects a predator, it emits a chirping sound as an alarm signal. When the alarm level exceeds a certain safety threshold, the explorer guides the followers to a safer foraging area.

(2) The sparrow's identity is defined by its ability to find better food, and while an individual's role may shift, the proportion of explorers to followers in the population remains constant.

(3) Sparrows with higher energy reserves act as explorers. To gain more energy, followers with lower energy may fly to other areas to forage.

(4) During foraging, followers always follow the explorers with higher energy reserves. In order to increase their own foraging success, some individuals may monitor the explorers and compete for more food resources.

(5) When a predator poses a significant threat, sparrows at the edges of the flock will quickly move toward safer areas for better positioning, while those in the middle of the group will move randomly.

In SSA, the best individual in the population will prioritize food acquisition during the search process. As explorers, they have a larger foraging range than the followers. During each iteration, the position update of the explorers is as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^{t} \cdot \exp\left(\frac{-i}{\alpha \cdot iter_{max}}\right) & \text{if } R_2 < ST \\ X_{i,j}^{t} + Q \cdot L & \text{if } R_2 > ST \end{cases} \quad (7)$$

Where, $X_{i,j}$ represents the position of the sparrow individual, where i is the current iteration number, and $iter_{max}$ is the maximum number of iterations. α is a random number within the range [0, 1]; R_2 ($R_2 \in [0, 1]$) and ST (ST $\in$ [0.5, 1]) represent the alarm value and safety threshold, respectively. Q is a random number that follows a normal distribution; L is a 1×d matrix where each element is 1.

When R_2 <ST, it indicates that there are no predators nearby, and the explorer can perform a global search. If R_2 ≥ST, this means that some sparrows have detected a predator, and all sparrows must take appropriate action. During foraging, some followers continuously monitor the explorers. If the explorer finds better food, the followers will immediately leave their current position to compete for the food. If they win the competition, they can immediately obtain the food; otherwise, they must continue to follow Rule 4. The position update for the followers is as follows:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{worst}^{t} - X_{i,j}^{t}}{\alpha \cdot iter_{max}}\right) & \text{if } i < n/2 \\ X_{p}^{t+1} + \left|X_{i,j}^{t} - X_{P}^{t+1}\right| \cdot A^{+} \cdot L & \text{otherwise} \end{cases} \quad (8)$$

In this equation, X_P represents the position of the optimal explorer, andX_{worst}^{t}denotes the current global worst position. n is the population size. A is a 1×d matrix, where each element is randomly assigned a value of either 1 or -1. The definition of A+ is as follows:

$$A^{+} = A^{T}(AA^{T})^{-1} \quad (9)$$

When i >n/2, it indicates that the fitness value of the i -th follower is relatively low, and its state is poor, requiring it to fly to other areas for foraging. Among the population, 10% to 20% of the individuals will recognize the danger, and the initial positions of these individuals are randomly generated within the population.

$$X_{i,j}^{t+1} = \begin{cases} X_{best}^{t} + \beta \cdot \left|X_{i,j}^{t} - X_{best}^{t}\right| & \text{if } f_i > f_g \\ X_{i,j}^{t} + K \cdot \left(\frac{\left|X_{i,j}^{t} - X_{worst}^{t}\right|}{(f_i - f_W) + \varepsilon}\right) & \text{if } f_i = f_g \end{cases} \quad (10)$$

Through the above model, by controlling parameters such as the number of sparrows and the predator-follower ratio, a complete Sparrow Search Algorithm (SSA) is formed. The SSA excels in both convergence speed and accuracy, primarily due to the unique guidance methods for predators and followers. Additionally, SSA possesses a certain degree of global search capability, allowing it to quickly escape from local optima.

3.3. Sparrow Search Algorithm Optimized Long Short-Term Memory Network

By combining the two algorithms, the LSTM (Long Short-Term Memory) network leverages its ability to predict time-series data, particularly for journey time predictions. The parameters in LSTM are optimized through the Sparrow Search Algorithm (SSA), which searches for the optimal solution within a reasonable range. This integration results in the SSA-LSTM model. The specific steps are as follows:

Data Processing: Process the raw data to form time-series journey time data.

Data Normalization: Standardize the raw data to have zero mean and unit variance. Use the first 90% of the data as the training set and the last 10% as the test set.

LSTM Model Creation: Create an LSTM neural network with a single input and a single output model. Define the SSA optimization algorithm, setting the number of sparrows in the population, the number of iterations, and other basic parameters. Select four groups of data from the LSTM model—hidden units, maximum training epochs, initial learning rate, and L2 parameters—to serve as the "sparrow position information" in SSA.

SSA Algorithm Execution: Begin the SSA algorithm after setting parameters such as the alarm value (ST), searcher ratio (PD), and the proportion of individuals aware of danger (SD). Initially, initialize the sparrow positions and calculate their initial fitness. In the SSA-LSTM model, root mean square error (RMSE) is used as the fitness evaluation index. Then, through the process described in Section 3.2, update the positions of sparrows with different roles and continue iterating until the highest fitness sparrow position is found, which leads to the optimal solution for the overall model.

Model Prediction: Finally, use the best parameters obtained from the SSA search to input into the LSTM model and make predictions for the data.

The specific workflow is illustrated in Figure 4.

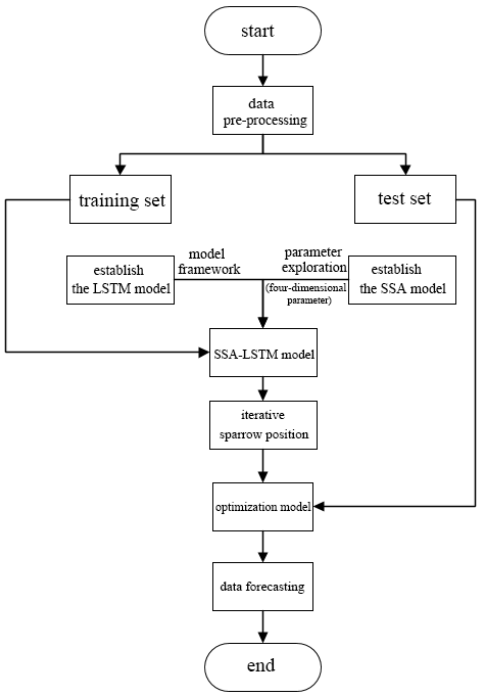

Figure 4. Illustrates the prediction process of the SSA-LSTM model.

4. Travel Time Case Study

4.1. Data Analysis

Using vehicle entry and exit records from the ETC system on the Jin-Gang Expressway, the travel time for each vehicle was calculated by determining the difference between its entry and exit times. Subsequently, data cleaning and processing were performed. The data was then segmented into 60-minute intervals, and the average travel time for each interval was calculated. This analysis focused on the average travel time between the Tianjin Toll Station and the Dagang Toll Station on the Jin-Gang Expressway in August 2018. Figure 5 presents a sample of the raw data, while Figure 6 shows a sample of the processed data.

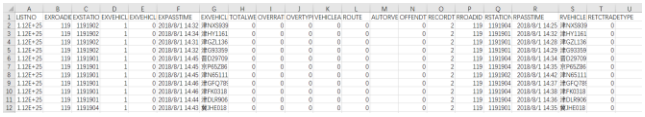

Figure 5. Sample of Raw Data

time	TT
2018/8/6 7:00	528.81
2018/8/6 7:30	530.53
2018/8/6 8:00	509.81
2018/8/6 8:30	518.69
2018/8/6 9:00	524.99
2018/8/6 9:30	540.58
2018/8/6 10:00	530.96
2018/8/6 10:30	564.62
2018/8/6 11:00	531.03
2018/8/6 11:30	515.74
2018/8/6 12:00	534.83
2018/8/6 12:30	522.6
2018/8/6 13:00	500.08

Figure 6. Sample of Processed Data

Building on the data processing described above, ten weekdays were selected as the dataset: August 6–10 and August 13–17. This selection ensures the data shares similar characteristics due to being weekdays while providing a sufficient data volume for model training. Additionally, nighttime data from 11:00 PM to 7:00 AM was excluded, as vehicle volumes during these hours are very low, leading to larger data errors and limited predictive significance. The final processed travel time data is shown in Figure 7.

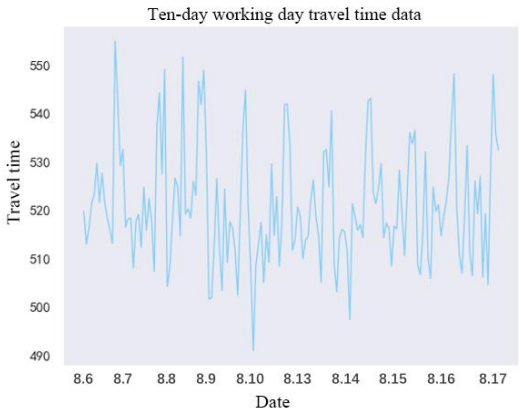

Figure 7. Travel Time Data for Ten Weekdays

The data reveals two daily peaks during weekdays, where travel time values are significantly higher. A detailed analysis indicates that these peaks occur in the morning and afternoon, corresponding to the morning and evening rush hours. This suggests that the data exhibits periodicity and the fundamental characteristics of time series data. Therefore, the LSTM model can be employed for prediction, along with the improved SSA-LSTM model for enhanced forecasting.

4.2. SSA-LSTM Model Prediction

SSA-LSTM data was used to predict the above data. In practice, we allocated 90% of the data for training and the remaining 10% for testing. This involved using data from 9 days to train the model and then predicting the data for the 10th day. During the forecasting process, we not only employed the SSA-LSTM model but also incorporated an LSTM model with fine-tuned parameters for comparative analysis. The comparative results are presented in Figures 8-9, where the blue line represents the predictions from the LSTM model, and the red line indicates the predictions from the SSA-LSTM model.

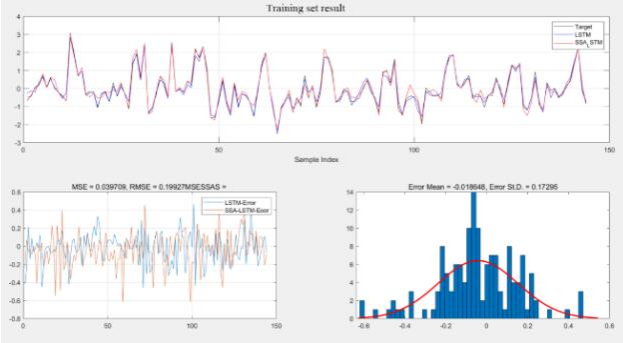

Figure 8. Results of the Training Dataset

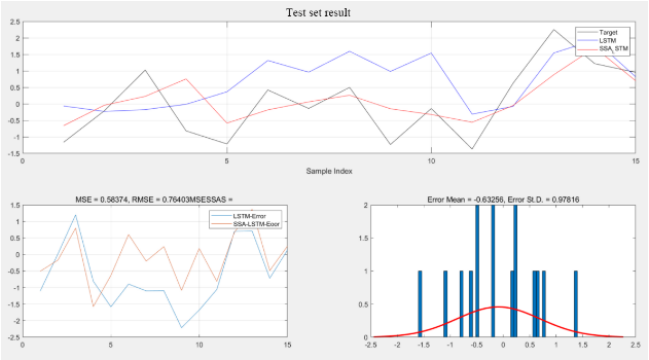

Figure 9. Results of the Test Dataset

The SSA-LSTM model achieved a training set MSE of 0.0397 and a test set MSE of 0.584. In comparison, the original LSTM model had a training set MSE of 0.0301 and a test set MSE of 1.293. This indicates that while the SSA-LSTM model showed a 24.18% deterioration in training set MSE, it achieved a 121.40% improvement in test set MSE, resulting in a significant optimization. The SSA-LSTM model, by more effectively searching for optimal parameters, experienced a slight decrease in training set MSE but doubled the prediction accuracy of the original LSTM model for travel time.

Regarding the convergence curve of the SSA-LSTM model, as shown in Figure 10, the SSA model found the optimal solution after just 12 iterations, reducing the fitness function to below 0.65. The final optimized parameters were 34 hidden units, a maximum of 76 training cycles, an initial learning rate of 0.1, and an L2 parameter of 0.001. The relatively small values for the learning rate and L2 parameter suggest that their search range was limited, while the hidden units and training cycles differ notably from those of the original LSTM model. The high accuracy of the SSA-LSTM model on the test set demonstrates the search optimization capability of SSA, which can effectively be combined with the LSTM model.

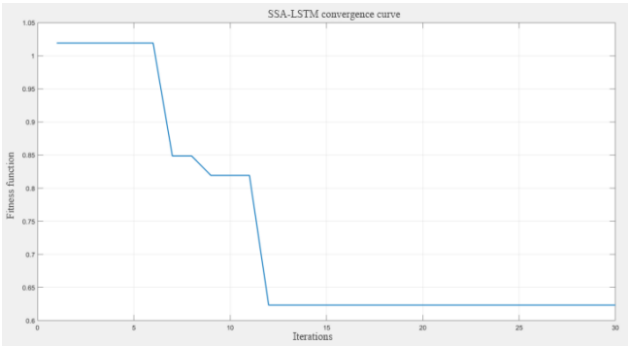

Figure 10. Convergence Curve of SSA-LSTM

5. Conclusion

The SSA search optimization algorithm, when combined with the LSTM model, forms a new SSA-LSTM model that improves prediction accuracy by approximately 120%. Additionally, the model requires relatively few iterations, achieving the optimal solution after around 12 iterations without consuming excessive computational power or resources. However, when integrated with LSTM, the SSA model primarily focuses on optimizing the selection of LSTM model parameters—by searching for more reasonable parameters to enhance prediction accuracy. This improvement raises the lower bound of the model, making it more efficient in finding optimal parameters quickly, but it does not extend the model's upper bound. Therefore, the applicability of the SSA-LSTM model is similar to that of the LSTM model, with the key advantage being faster identification of optimal parameters. In this study, the introduction of the Sparrow Search Optimization algorithm to optimize network parameters significantly enhanced the prediction accuracy of the original model. However, it primarily improved the model's lower bound and did not address the issue of a low upper bound. To further increase accuracy, additional methods may need to be explored. Despite its limitations, the application of the Sparrow Search Optimization algorithm in model parameter optimization is highly successful, as it quickly optimizes the parameters of the original model, leading to a significant improvement in overall model performance.

References

[1] P. Nelson and P. Palacharla, “A Neural network model for datafusion in ADVANCE,” Proceedings of the Pacific Rim Trans Tech Conference .1993, 237-293.

[2] A. Ladino, A. Kibangou , D. C. C. Wit, et al, A real time forecasting tool for dynamic travel time from clustered time series.Transportation Research Part C, 2017, 80216-238.

[3] T. Cristóbal, G. Padrón, A. Quesada-Arencibia, et al, Bus Travel Time Prediction Model Based on Profile Similarity. Sensors, 2019, 19(13):2869- 2869.

[4] X. Zhan, V. S. Ukkusuri and C. Yang, A Bayesian mixture model for short-term average link travel time estimation using large-scale limited information trip-based data. Automation in Construction, 2016, 72237-246.

[5] A. Bezuglov and G. Comert, Short-term freeway traffic parameter prediction: Application of grey system theory models. Expert Systems With Applications, 2016, 62284-292.

[6] Y. Duan, L. V. Yisheng and F. Y. Wang, Travel time prediction with LSTM neural network//2016 IEEE 19th international conference on intelligent transportation systems (ITSC). IEEE, 2016: 1053-1058.

[7] M. Li and Y. L. Gu, A Forecasting Model for Shortterm TravelTime on Freeways under Rainfall Conditions.Journal of Transport Information and Safety, 2018, 36(04):90-96.

[8] J. Zhang and J. Sun, Prediction of Urban Expressway Travel Time Based on SVM. Journal of Transportation Systems Engineering and Information Technology, 2011, 11(02):174-179.

[9] S. J. Li and J. F. Song, Travel Time Prediction of Freeway Based on Clustering Analysis, Computer Simulation, 2019, 36(02):384-389.

[10] W. Zhang, R. Li and Z. Xie, Travel Time Prediction of Urban Road Based on Deep Learning. Journal of System Simulation，2017, 29(10):2309-2315.

[11] G. Mustafa, Y. Hwang and S. J. Cho, Predicting Bus Travel Time in Cheonan City Through Deep Learning Utilizing Digital Tachograph Data. Electronics, 2024, 13(9): 1771.

[12] H. Li, Z. Wang, X. Li, H. Wang, Y. Man and J. Shi, Travel Time Probability Prediction Based on Constrained LSTM Quantile Regression, Journal of Advanced Transportation, vol. 2023.

[13] X. Jia, W. Zhou and H. Yang, Short-term traffic travel time forecasting using ensemble approach based on long short-term memory networks, IET Intelligent Transport Systems, vol. 2023.