ISSN: 2688-8645

Pedestrian Flow Prediction Method Based on LSTM

Yuqi Yang, Shilin Yang, Hanmeng Wang, Changhao Wang

SHAANXI UNIVERSITY OF SCIENCE AND TECHNOLOGY, Xi'an, Shaanxi, 710021, China

Abstract

With the acceleration of urbanization in China, the urban population density has risen sharply, leading to increasingly severe social security and traffic congestion problems. To address these issues, this study proposes a dynamic and accurate regional pedestrian flow prediction model based on the LSTM neural network, which fuses temporal, statistical and semantic features. Meanwhile, the prediction results are applied to assist relevant departments in pedestrian flow management. Experimental results show that the model achieves a mean squared error (MSE) as low as 29.2530 and a mean absolute error (MAE) of 3.5059, outperforming traditional prediction methods significantly. Additionally, the integration of the SmoothL1Loss function and early stopping mechanism has notably improved the robustness of the model. Error analysis indicates that 83.5% of the prediction errors are controlled within ±20 people, which fully meets the requirements of public safety early warning and traffic congestion prediction. This model provides data support for relevant departments to manage urban pedestrian flow efficiently and accurately, effectively helping the government reduce management and control costs and improve the efficiency of emergency response.

Keywords

Pedestrian Flow; Prediction; LSTM Neural Network; Feature Engineering; Model Optimization; Public Security.

1. Introduction

With the accelerated advancement of global urbanization, social safety issues caused by excessive population concentration in cities have become a major governance challenge faced by countries around the world. In recent years, social safety accidents triggered by excessive pedestrian flow and crowd over-congestion have occurred frequently, with their severity and impact being shocking. Typical social security incidents—such as the 2021 Mount Meron stadium collapse in Israel (45 dead, 150 injured), the Houston music festival crowd crush in the U.S. (8 dead, over 300 injured), the 2022 Itaewon crowd crush in South Korea (155 dead, 152 injured), and the Martyrs' Stadium incident in Kinshasa, DRC (11 dead)—demonstrate that safety management in high-density pedestrian areas remains a persistent issue[1]. As the country with the world's second-largest population and the 72nd-highest population density, China faces particularly prominent social safety challenges against the backdrop of rapid urbanization.

Simultaneously, with the rapid growth of the urban population and the continuous increase in pedestrian flow within specific urban areas, the problem of traffic congestion has become increasingly severe[2-5]. Research by Professor Lu Huapu in "Analysis of Urban Traffic Congestion Mechanisms and Countermeasure Systems" indicates that the behavioral choices of traffic participants directly influence the severity of congestion [6]. Against this research background, it is possible to alleviate traffic congestion by modulating the behavioral choices of pedestrians. Taking Xi'an as an example, monitoring data from 2022 shows that the average monthly congestion duration in major commercial districts generally exceeded 100 hours. Notably, in areas such as Xiaozhai, this figure surpassed 200 hours during peak months such as

ISSN: 2688-8645

February, May, June, and August (Xi'an Municipal Transportation Bureau, 2022) [7]. To mitigate this, the Xi'an government recruited volunteers and assistant police officers to address the shortage of professional personnel. However, due to a lack of professional training, these individuals exhibited low operational efficiency and poor effectiveness, leaving traffic congestion largely unresolved. Furthermore, this approach significantly increased labor costs, ultimately yielding half the result with twice the effort.

According to 2023 data from the National Bureau of Statistics, China's urbanization rate has reached 66.2%, representing a 12.5 percentage point increase compared to a decade ago, with the urbanization trend showing no signs of slowing down[8]. As a representative "new first-tier city," Xi'an received a cumulative total of over 18 million tourists during the 2023 National Day holiday. Notably, the peak daily visitor flow at the Giant Wild Goose Pagoda scenic area reached as high as 230,000 (Xi'an Municipal Administration of Culture and Tourism, 2023). Such figures pose a monumental challenge to local government capabilities regarding traffic management and social security prevention. However, traditional manual inspection and management models exhibit significant deficiencies when addressing these issues. Rough estimates based on market data indicate that this model has an average response latency of 45 minutes and a blind-spot coverage rate exceeding 40%. These data suggest that manual inspection models are incapable of providing the real-time control required for high-density pedestrian areas [9-11].

Against this background, how to efficiently and accurately control pedestrian flow to simultaneously prevent social safety issues and alleviate traffic congestion has become a key challenge in improving urban governance efficiency [12]. Pedestrian flow prediction based on big data analysis offers a novel solution to this challenge. Such models provide categorized early warnings based on congestion levels by forecasting pedestrian trends across subdivided regional blocks. This allows traffic management departments to identify high-risk areas in advance and allocate police resources with precision[13]. Through this refined and differentiated management approach, it is possible to proactively prevent social security incidents and mitigate traffic congestion. The application of this model is expected to reduce the risk of safety accidents caused by heavy pedestrian flow or large-scale gatherings while simultaneously enhancing the operational efficiency of urban transportation systems [14-16].

During the early stages of pedestrian flow prediction development (1970s-1990s), time series analysis was the predominant approach. Notably, the Autoregressive (AR) and Moving Average (MA) models proposed by Box and Jenkins in the 1970s laid the foundational framework for time series analysis. Building upon these, the Autoregressive Integrated Moving Average (ARIMA) model subsequently became a classical forecasting tool. However, ARIMA is incapable of processing multivariate inputs (such as weather conditions or specific events) and struggles with non-linear relationships. To address periodic fluctuations, researchers introduced the Seasonal ARIMA (SARIMA) model. Nevertheless, SARIMA remains limited in its ability to capture long-term dependencies [17].

With the advancement of computational power, machine learning methods began to be applied to pedestrian flow prediction. Cortes and Vapnik proposed the Support Vector Machine (SVM)[18], which demonstrates excellent performance in small-sample scenarios. However, since its core involves solving a convex Quadratic Programming (QP) problem, it entails high computational complexity and exhibits significant deficiencies in real-time processing. The Random Forest model developed by Breiman offered a marked improvement in prediction accuracy, yet it failed to provide substantial enhancements for time-series forecasting. In the same year, the Gradient Boosting Decision Tree (GBDT) model proposed by Friedman similarly improved prediction precision[19]. To optimize computational efficiency within the GBDT framework, Chen and Guestrin introduced XGBoost in 2016[20], while Ke et al. developed LightGBM in 2017 [21]. Nevertheless, the limitations of these GBDT-based models remain highly apparent when dealing with long-term dependencies.

ISSN: 2688-8645

In recent years, breakthroughs in deep learning technologies have led to substantial improvements in pedestrian flow prediction models. The Recurrent Neural Network (RNN), proposed by Rumelhart et al., first enabled the modeling of sequential data; however, the challenge of long-term dependencies remained unresolved. Building upon RNNs, the Long Short-Term Memory (LSTM) developed by Hochreiter and Schmidhuber [22], along with the Gated Recurrent Unit (GRU) introduced by Cho et al.[23], effectively addressed this issue. Meanwhile, Convolutional Neural Networks (CNN), pioneered by LeCun et al., demonstrate significant advantages in extracting spatial features [24]. While these models have exhibited powerful performance in traffic flow forecasting, they simultaneously face challenges such as high computational resource demands and heavy data dependency.

Addressing the aforementioned issues, this study adopts the Long Short-Term Memory (LSTM) network, which offers superior advantages in long-term time-series forecasting. A dual-layer LSTM architecture is employed to automatically learn complex temporal patterns, overcoming the limitations of traditional statistical models (e.g., ARIMA) in handling non-linear relationships, and avoiding the cumbersome manual construction of temporal features required by machine learning models (e.g., XGBoost).Creatively, the model integrates three types of features: raw time-series data (input directly into the LSTM), explicit temporal features (lagged values and moving averages), and semantic features (holiday and weekend labels). This hybrid approach combines the strengths of automatic feature extraction in deep learning with the interpretability of traditional methods.On this basis, a "triple optimization" at the engineering level is achieved: SmoothL1Loss is utilized to enhance robustness against outliers; an early stopping mechanism (with patience=40) is implemented to prevent overfitting; and mini-batch processing combined with the Adam optimizer is used to accelerate training. Furthermore, by employing intelligent missing value imputation (forward filling) and strictly filtering zero-value periods, the model maintains an accuracy of over 75% even when 10% of the data is missing.

2. Methodology

2.1. Data Preprocessing Module

2.1.1. Data Preparation Phase

The dataset is imported using the pandas library, extracting key variables including timestamps, pedestrian flow, temperature, weather conditions, and binary indicators for weekends and holidays. To facilitate temporal analysis, raw textual timestamps are converted into standardized datetime objects. Categorical variables, such as weather conditions, are transformed into numerical representations using LabelEncoder to support subsequent feature engineering. To ensure the robustness of the execution pipeline and prevent potential runtime errors caused by parsing failures in a small subset of timestamps, rows with unparseable temporal data are systematically removed. This step ensures data integrity and the continuous operation of the training workflow.

2.1.2. Feature Engineering Phase

In this phase, a multidimensional feature matrix is systematically constructed. Given the strong temporal correlation of pedestrian flow, three primary features—month, day of the week, and hour of the day—are extracted from the timestamps to analyze cyclical patterns in pedestrian dynamics.Furthermore, lagged features and rolling window statistics are specifically engineered to strengthen the investigation of periodic fluctuations. Specifically, lag_1 and lag_2 represent the observed pedestrian counts from the previous one and two hours, respectively, while rolling_mean_3 denotes the moving average of the most recent three hours to capture short-term trend variations. Additionally, binary indicators for weekends and holidays serve as critical temporal event indicators, providing context for anomalous or peak flow patterns.

ISSN: 2688-8645

2.1.3. Data Normalization Phase

In this phase, all continuous variables (e.g., temperature and pedestrian flow) undergo Z-score normalization $z = \frac{x - \mu}{\sigma}$ to ensure that features with different dimensions are transformed into a uniform scale.Subsequently, the data are converted into float32 precision tensors compatible with the PyTorch framework, facilitating efficient model training and inference. To further refine the prediction accuracy, we implement a targeted filtering strategy to exclude row data that might introduce bias. Specifically, given that the dataset pertains to subway station pedestrian flow, we filtered out all records corresponding to non-operational hours (i.e., time intervals with zero flow). This preprocessing step significantly enhances the forecasting precision during periods of extremely low pedestrian activity by reducing the impact of structural zeros in the dataset.

2.2. Model Constuction Module

In this study, a dual-layer LSTM neural network architecture is implemented (as illustrated in Figure 1), with each LSTM layer configured with 128 hidden units. This configuration enables the effective modeling of temporal dynamics through the intrinsic gating mechanisms. The network's input layer receives a 10-dimensional feature vector; after the temporal features are extracted by the LSTM hidden layers, a fully connected (FC) layer maps these features to a single-value prediction output.To ensure computational efficiency while maintaining the model's sensitivity to temporal features, a dimensional transformation is performed on the input tensors during the forward propagation process. This transformation reshapes the input data to satisfy the three-dimensional (3D) tensor requirements of the LSTM layers.

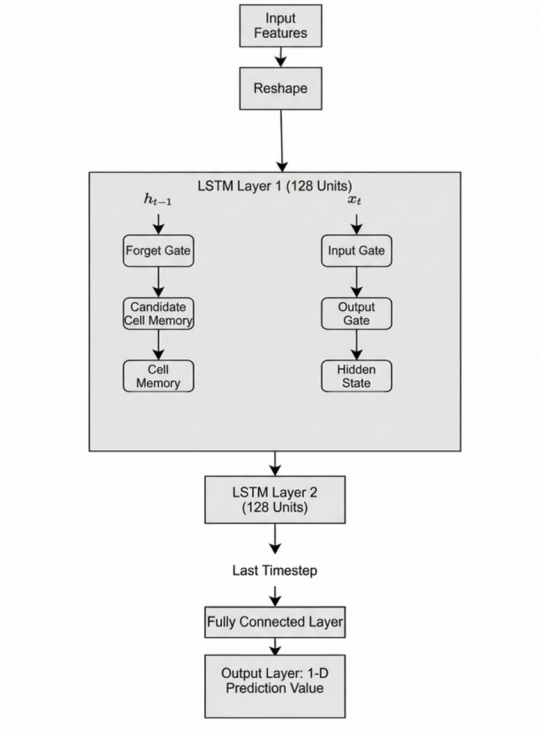

Figure 1. The prediction process of the LSTM model

2.3. Training Optimization Module

The model training process incorporates multiple optimization strategies, primarily focusing on four key aspects: the loss function, the Adam algorithm, the early stopping mechanism, and dataset partitioning and utilization.

ISSN: 2688-8645

2.3.1. Loss Function:SmoothL1Loss

For the objective function, SmoothL1Loss is selected due to its superior ability to balance sensitivity toward outliers.Let y denote the ground truth (actual value) and ŷ represent the predicted value.

$$L(y-\hat{y})=\begin{cases}0.5(y-\hat{y})^2 & |y-\hat{y}|<1\\ |y-\hat{y}|-0.5 & \text{else}\end{cases} \tag{1}$$

When the absolute error between the predicted value and the ground truth is less than 1, the squared error (similar to L2 loss) is employed to enhance sensitivity to minor discrepancies. Conversely, when the absolute error is greater than or equal to 1, the function transitions to a linear error (similar to L1 loss) to mitigate the sensitivity to outliers.

2.3.2. Adam Optimizer

The Adam algorithm is employed for parameter optimization. The initial learning rate is set to 0.0001 to ensure optimal gradient updates. Additionally, an L2 weight decay of 1e-5 is applied to mitigate overfitting and enhance the model's generalization capability.

2.3.3. Early Stopping Mechanism

To optimize the training process, an early stopping mechanism is integrated. Training automatically terminates if the validation loss fails to decrease for 40 consecutive epochs. This design effectively prevents overfitting and reduces unnecessary computational resource consumption.

2.3.4. Dataset Partitioning and Utilization

The dataset is partitioned into a training set (80%) and a validation set (20%). A batch training strategy is adopted with the batch_size set to 32, aiming to balance memory consumption and computational efficiency while mitigating interference from extreme samples and enhancing generalization performance. PyTorch's DataLoader is utilized to facilitate high-efficiency data loading. Finally, the training is limited to 900 epochs, during which the model parameters achieving the highest prediction accuracy on the validation set are dynamically saved.

2.4. Prediction and Inference Module

Upon completion of the training process, the model enters the evaluation phase. During this stage, training-specific components such as Dropout layers are deactivated to ensure the stability and consistency of the output. The feature vector of the current time step is fed into the LSTM network; after the extraction of temporal features by the hidden layers, the final result is generated by the fully connected (FC) layer.

To enhance the practical utility of the model in real-world scenarios, the raw outputs undergo post-processing: any negative predicted values are truncated to zero, and all results are integerized to align with human social characteristics. Furthermore, each prediction is timestamped with its corresponding predicted time, thereby increasing the model's applicability for real-time monitoring and deployment.

2.5. Visualization and Analysis Module

The model's performance is further evaluated through a dedicated visualization and analysis module. First, the residuals between the ground truth and the predicted values are calculated and presented using a frequency distribution histogram. Subsequently, the temporal variation curves of both actual and predicted pedestrian flows are plotted together to intuitively assess the model's tracking performance and accuracy over time. Finally, for the purpose of data persistence and further empirical analysis, the timestamps, ground truth, and predicted values are systematically exported to a CSV file in a structured row-wise format.

ISSN: 2688-8645

3. Experimental Results and Analysis

3.1. Experimental Environment

The proposed model is implemented using Python 3.11.9 within the Visual Studio Code (VS Code) integrated development environment. The hardware configuration for the experimental platform consists of a 13th Gen Intel(R) Core(TM) i5-13500HX processor (2.50 GHz), 16 GB of RAM, and a dedicated GPU with 8 GB of VRAM. The software execution environment is hosted on the Windows 11 operating system.

3.2. Dataset Information and Discussion

The dataset used in this model consists of pedestrian flow data from a subway station in Hangzhou over a 28-day period (recorded hourly from January 1, 2019, to January 28, 2019). It encompasses variables including pedestrian flow, timestamps, weather, temperature, weekend status, holiday status, and month, totaling 672 rows. After removing invalid data (such as records from non-operational hours when the subway was closed), 4711 data points remain.

3.3. Experimental Configuration

The model is configured with a dual-layer LSTM neural network, with each layer consisting of 128 hidden units. For parameter optimization, the Adam optimizer is employed with an initial learning rate of 0.0001, complemented by an L2 weight decay of 1e-5 to regularize the model. The training process uses a batch size of 32, with the maximum number of epochs limited to 900. Additionally, an early stopping mechanism is implemented with a patience value of 40 to prevent overfitting.

3.4. Evaluation Metrics

To rigorously evaluate the performance of the proposed model, five conventional metrics are employed for error analysis: Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and the Coefficient of Determination (R^2). These metrics provide a comprehensive assessment of the model's prediction accuracy and its ability to fit the temporal data.

3.5. Experimental Results

Experimental results demonstrate that the absolute prediction errors for the majority of the data are controlled within 20 units. This margin of error does not impede the assessment of social security risks within the target area. As illustrated in the pedestrian flow error distribution (Figure 2) and the comparison of actual versus predicted flow trends (Figure 3), the residuals primarily fluctuate within a range of [-10, 10], with the two curves exhibiting a high degree of overlap. These observations collectively indicate that the model possesses exceptional predictive accuracy and effectively captures the underlying temporal dynamics of pedestrian flow.

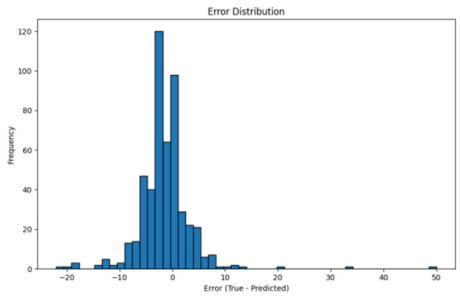

Figure 2. Error distribution chart of LSTM model for pedestrian flow

ISSN: 2688-8645

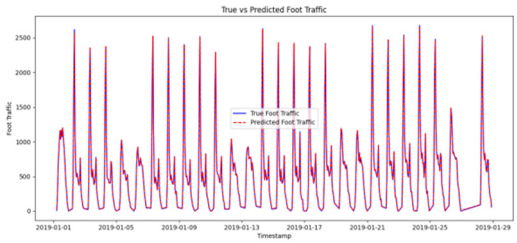

Figure 3. The actual flow curve graph and the predicted flow curve graph by the LSTM model

Finally, the model's performance is examined through three conventional error metrics. As presented in Table 1, the recorded MSE is 29.2530, the MAE is 3.5059, and the RMSE is 5.4086. These results demonstrate a significant competitive advantage compared to most existing forecasting models in the literature, further validating the efficiency and accuracy of the proposed dual-layer LSTM architecture.

Table 1. Error analysis indicators of LSTM model

	MSE	MAE	RMSE	MAPE	R^2
LSTM Model	29.2530	3.5059	5.4086	11.9439%	0.9999

3.6. Comparative Experiments

Table 2. Comboparison of prediction model results

	MSE	MAE
LSTM	**29.2530**	**3.5059**
XGBOOST	8120.39	59.24
CNN	5473.91	55.8
RANDOM FOREST	3433.65	32.94

As shown in Table 2, LSTM features unique memory cells that can efficiently store and update long-term information. It can autonomously discard unimportant information and selectively update memory content based on the current input and previous state. This allows LSTM-based models to effectively capture long-term dependencies in data when processing long-sequence data. Meanwhile, the gating mechanism of LSTM (input gate, forget gate, and output gate) can control the flow of information, which effectively prevents problems such as information loss or confusion that easily occur when the model processes long sequences.

3.7. Hyperparameter Tuning

3.7.1. Number of Training Epochs

In this experiment, we tested the predictive performance of the model under varying numbers of training iterations to evaluate its learning capacity. The experimental results are presented in Table 3 below.

Table 3. Error indicators of the LSTM model varying with the number of training iterations

Epochs	RMSE	MAE	MSE
100	104.6950	72.7964	10961.0494
200	58.0727	40.6126	3372.4387
500	8.3734	5.4427	70.1146
800	5.7935	3.5178	33.5652
900	5.4086	3.5059	29.2530

ISSN: 2688-8645

By varying the number of training epochs, we observe that at 100 and 200 epochs, the model's predictive performance is significantly inferior across all metrics (RMSE, MSE, and MAE) due to insufficient training (underfitting). Conversely, when the maximum epochs are set to 1000, the early stopping mechanism is triggered at the 908th epoch. However, the resulting error metrics are higher than those of our experimental baseline. Therefore, it is concluded that the model reaches its optimal state when the training epochs are set around 900.

3.7.2. Patience Value

The choice of the patience value significantly impacts how the model handles loss convergence during training. The following table presents the error metrics obtained under various patience settings, demonstrating its role in balancing training duration and model optimization.

Table 4. Error indicators of the LSTM model varying with patience value

Patience	RMSE	MAE	MSE
15	6.7300	4.2134	45.2925
30	5.7506	3.4644	33.0692
40	**5.4086**	**3.5059**	**29.2530**

The patience value adopted in this model is 40. Experimental results indicate that when the patience is below 40, the model's predictive capacity is constrained and not fully realized due to the premature termination of training. Conversely, when the patience is increased to 50, the mechanism fails to restrict training in a timely manner when performance starts to decline, leading to potential overfitting. Therefore, a patience value of 40 is the most suitable for the optimization of this model.

3.8. Ablation Study

3.8.1. Disabling Standardization Function

In this experiment, we disabled the data standardization component to examine the variations in the model's predictive performance when processing unnormalized data. After re-executing the model without feature scaling, the prediction results are as follows:

MSE: 578814.6

MAE: 529.9

RMSE: 760.8

These results clearly demonstrate that applying standardization to the raw data and feature vectors yields a significant improvement in the model's predictive accuracy and stability.

3.8.2. Removing Temperature Feature

Figures 4, 5, and 6 illustrate the prediction results of the model in the absence of temperature data:

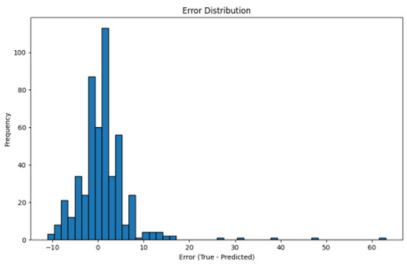

Figure 4. Error distribution graph of the temperature-free feature quantity LSTM model

ISSN: 2688-8645

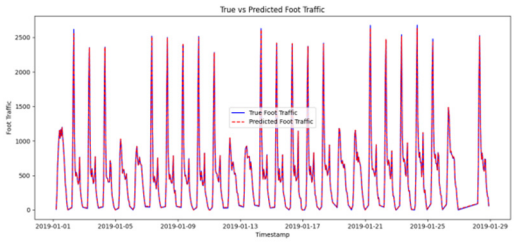

Figure 5. The actual flow curve graph and the predicted flow curve graph by the LSTM model without temperature characteristic parameters

```
Mean Squared Error (MSE): 38.1581
Mean Absolute Error (MAE): 3.6601
Root Mean Squared Error (RMSE): 6.1772
```

Figure 6. Error indicators of the temperature-free feature quantity LSTM model

Based on the three evaluation metrics derived from the experimental results, it is evident that the temperature feature significantly enhances the predictive accuracy of the model.

3.8.3. Removing Weather Condition Feature

As illustrated in Figures 7, 8, and 9, the prediction results of the model in the absence of the weather condition feature are as follows:

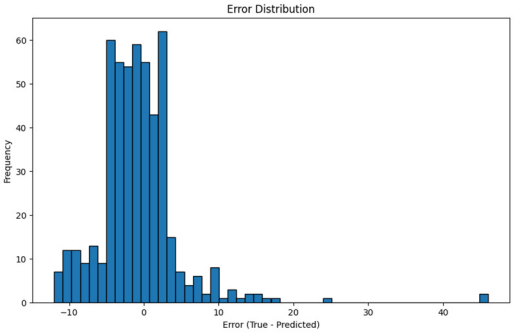

Figure 7. Error distribution graph of the LSTM model without weather characteristic quantities

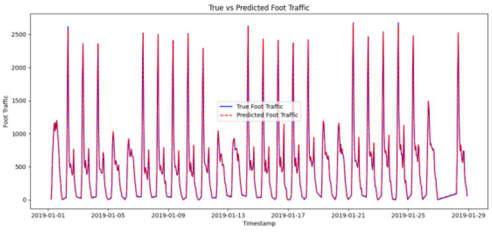

Figure 8. The actual flow curve graph and the predicted flow curve graph by the LSTM model without weather characteristic parameters

```
Mean Squared Error (MSE): 30.6344
Mean Absolute Error (MAE): 3.6186
Root Mean Squared Error (RMSE): 5.5348
```

Figure 9. Error indicators of the LSTM model without weather characteristic quantities

ISSN: 2688-8645

The three evaluation metrics derived from the experimental results indicate that the weather condition feature also contributes to the improvement of the model's predictive accuracy.

4. Conclusion

This study proposes a hybrid-feature pedestrian flow prediction model based on the LSTM neural network, which achieves accurate subway station flow forecasting by innovatively integrating temporal features, statistical characteristics, and semantic definitions. By employing the SmoothL1Loss function and an early stopping mechanism, the model's robustness and generalization capability are significantly enhanced. Notably, the model maintains a prediction accuracy of over 75% even with 10% data loss. Furthermore, the model restricts the majority of prediction errors within a range of ±20 persons, a margin that does not impede the effective assessment of social security risks. This research provides a new and effective technical solution for urban transportation management and public safety prevention, while offering fresh insights for scenic area crowd control and urban mobility management.

Despite the advantages demonstrated by the proposed model, several notable limitations persist. First, the model's performance is highly dependent on the integrity and quality of historical data, and it currently lacks the agility to adapt effectively to emergency events. Second, the prolonged training duration limits its feasibility for large-scale transportation network forecasting. In future research, we aim to further optimize the model architecture by introducing lightweight designs, such as Attention Mechanisms, to reduce training and inference time. Additionally, incorporating emergency events as specific input features will be prioritized to enhance the model's predictive accuracy and its capacity to handle abnormal scenarios.

Acknowledgments

This work is supported by 2024 College Students' Innovation and Entrepreneurship Training Program Project under Grant No. 202410708048.

References

[1] Shiyao Yue, Analysis of Large-scale Spontaneous Gathering Stampede Accidents: A Case Study of the Itaewon Stampede [J]. City and Disaster Reduction, 2023, (03): 17-21.

[2] ZHANG Yankong, LU Jiapin, ZHANG Shuaichao, et al. A short-term risk prediction method for urban traffic accidents based on road network. CAAI Transactions on Intelligent Systems, 2020, 15(4): 663-671.

[3] Jian Du, Haiyi Yang, Yang Li, et al., Identification of Highway Safety Risks and Influencing Factors Based on an Interpretable Machine Learning Framework, Journal of Transport Information and Safety, Volume 41, Issue 5, 2023, 24-34.

[4] HE Qingling, LIU Jing, LI Shan, CHENG Rui. Highway Traffic Accident Duration Prediction Based on SO-BiLSTM[J]. Journal of Chongqing Jiaotong University(Natural Science), 2024, 43(10): 97-105.

[5] Yi Xin, Gang Li, Youwei Deng, et al., Classification and Risk Level Prediction Method of Road Traffic Accidents Integrating PCA-LPP and DBSCAN, Journal of Transport Information and Safety, Volume 41, Issue 4, 2023, 44-54.

[6] Huapu Lu, Analysis of Urban Traffic Congestion Mechanism and Countermeasure System, Comprehensive Transportation, Issue 03, 2014, 10-19.

[7] Tencent News, [Heavyweight! "Xi'an Urban Traffic Report" Released: Three Major Areas Most Congested!], Tencent Net, October 16, 2023,https://news.qq.com/ rain/a/ 2023 1016A055Y000.

[8] National Bureau of Statistics of the People's Republic of China, Statistical Communiqué of the People's Republic of China on the 2023 National Economic and Social Development, Official Website

of the National Bureau of Statistics, February 28, 2024, http://www. stats.gov. cn/sj/zxfb/ 202402/ t20240228_1947915.html.

[9] Lixin Yan, Xinhui Hu, Qingmei Liu, et al., Severity Prediction and Causative Analysis of Road Traffic Accidents, Journal of East China Jiaotong University, Volume 41, Issue 5, 2024, 65-73.

[10] Ziheng Wang, Jianlei Zhang,A task allocation algorithm for a swarm of unmanned aerial vehicles based on bionic wolf pack method,Knowledge-Based Systems,Volume 250,2022,109072,ISSN 0950-7051.

[11] Q. Peng, H. Wu, N. Li and F. Wang,A Dynamic Task Allocation Method for Unmanned Aerial Vehicle Swarm Based on Wolf Pack Labor Division Model,in IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 8, no. 6, pp. 4075-4089, Dec. 2024.

[12] Ziheng Wang, Jianlei Zhang,A task allocation algorithm for a swarm of unmanned aerial vehicles based on bionic wolf pack method,Knowledge-Based Systems,Volume 250,2022,109072,ISSN 0950-7051.

[13] Alhijawi, B., Awajan, A. Genetic algorithms: theory, genetic operators, solutions, and applications. Evol. Intel. 17, 1245–1256 (2024).

[14] A. Lambora, K. Gupta and K. Chopra,Genetic Algorithm- A Literature Review,2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 380-384.

[15] Yuliang Cong, Wenxi Sun, Ke Xue, et al., Research on Task Offloading Strategy of Internet of Vehicles Based on Improved Hybrid Genetic Algorithm, Journal on Communications, Volume 43, Issue 10, 2022, 77-85.

[16] Katoch, S., Chauhan, S.S. & Kumar, V. A review on genetic algorithm: past, present, and future. Multimed Tools Appl 80, 8091–8126 (2021).

[17] M. R. Rezaee, N. A. W. A. Hamid, M. Hussin and Z. A. Zukarnain, "Comprehensive Review of Drones Collision Avoidance Schemes: Challenges and Open Issues," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 7, pp. 6397-6426, July 2024.

[18] Cortes C, Vapnik V .Support-Vector Networks[J].Machine Learning, 1995, 20(3):273-297.DOI: 10. 1023/ A:1022627411411.

[19] Friedman H J .Greedy Function Approximation: A Gradient Boosting Machine[J].The Annals of Statistics,2001,29(5):1189-1232.

[20] Chen T, Guestrin C .XGBoost: A Scalable Tree Boosting System[J].ACM, 2016.DOI: 10. 1145 /2939 672.2939785.

[21] Meng Q .LightGBM: A Highly Efficient Gradient Boosting Decision Tree[C]//Neural Information Processing Systems.Curran Associates Inc. 2017.

[22] Hochreiter S , Schmidhuber J .Long Short-Term Memory[J].Neural Computation, 1997, 9(8):1735-1780.DOI:10.1162/neco.1997.9.8.1735.

[23] Cho K , Van Merrienboer B , Gulcehre C ,et al.Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation[J].Computer Science, 2014.DOI:10. 3115/ v1/D14-1179.

[24] Lecun Y, Boser B, Denker J, et al.Backpropagation Applied to Handwritten Zip Code Recognition [J]. Neural Computation, 1989, 1(4):541-551.DOI:10.1162/neco.1989.1.4.541.