Research on Short-Term Traffic Flow Prediction Based on KOA-CNN-BiGRU-MultiAttention Hybrid Neural Network Model

Yitian Huo *

Department of Transportation, Inner Mongolia University, Hohhot, China, 010030

* Corresponding Author Email: 18522870908@163.com

Abstract. The demand for transportation brought about by urban population expansion has been increasing in the last few decades, and since the construction of transportation infrastructure is becoming highly saturated, it is tougher for regular traffic management measures to effectively ease traffic congestion in daily travel. Due to the wide application of deep learning in the field of transportation, a great number of deep learning models have been applied to traffic flow prediction in a various traffic environments, and short-term traffic flow prediction is undoubtedly among the most economical and effective measures to assist traffic management. In this paper, a hybrid neural network of KOA-CNN-BiGRU-MultiAttention based on pytorch deep learning framework for short-term traffic flow prediction is proposed. This model shows significant advantages of high prediction accuracy and robustness compared with traditional CNN-BiGRU-Attention prediction model, and its rapid response has significant advantages in predicting the characteristics of road network traffic flow.

Keywords: KOA, CNN, Bidirectional GRU, Multi Attention, Short-term Traffic Flow Prediction.

1. Introduction

Considering the current saturation of urban transportation infrastructure, it is increasingly difficult for conventional traffic management measures to effectively alleviate congestion during peak hours. Short term traffic flow prediction, as a cost-effective and practical solution, plays a crucial role in intelligent transportation systems by enabling proactive traffic control, optimizing signal timing, and guiding route choices in real time. With the rapid development of deep learning, numerous data-driven models have been proposed to address the complexity and uncertainty inherent in traffic dynamics, offering new possibilities for accurate and timely forecasting.

Recent research has explored various spatial-temporal modeling strategies for enhancing prediction accuracy. Wang et al developed an LSTM-RNN model that reconstructs and strengthens time-series input to improve both accuracy and timeliness in short-term forecasting [1]. Feng et al introduced a multi-component spatial-temporal graph convolutional network (MCSTGCN) capable of capturing temporal dependencies across recent, daily and monthly patterns, thus improving spatial–temporal correlation learning [2]. Yuan et al proposed a dilation-causal convolutional neural network (DCFCN) to expand the receptive field and prevent information leakage, significantly boosting computational efficiency [3]. Nisha Singh et al. presented AST-Deep, an attention-based deep learning model aimed at improving the accuracy and reliability of short-term traffic flow forecasting, which achieves a 1 to 5 % improvement in MAE and RMSE over the best-performing baseline model as the prediction horizon increases [4]. Sonia Mrad et al used wavelet transform (WT) to handle with the non-stationary characteristics of traffic flow data, which is applied to signal decomposition for the elimination of redundant data from input matrices, and has produced lower loss for all step-horizons analyzed comparing with existing prediction methods [5]. Naheliya Bharti et al applied a Chaotic Particle Swarm Optimization (CPSO) technique to a two-layer bidirectional LSTM memory network, which significantly accelerates the convergence of the Particle Swarm Optimization (PSO) algorithm and improves forecast accuracy of traffic speed [6]. Guowen Dai et al. introduced a personalized lightweight federated learning framework (PLFL), which is capable of collaboratively training a unified global traffic flow prediction model without compromising the privacy of individual datasets [7]. Xiaoqing Wang et al proposed a short-term traffic prediction method based on vehicle trip chain features, which uses a pre-built CNNs-LSTM model to improve prediction accuracy and shorten time consumption [8]. In order to solve the challenge posed by the inherent nonlinearity and stochasticity of traffic flow prediction, Ali Reza Sattarzadeh et al proposed an attention mechanism that uses multi-layered hybrid architectures to extract spatial–temporal and nonlinear characteristics [9]. To determine a reasonable spatial-temporal correlation range, Lingjuan Chen et al constructed an Inverse Isochrone (ISOv) model that considers the dynamic diffusion time and direction of traffic flow, experimental results on a real data set show that the complete model improves prediction accuracy by approximately 15% in terms of RMSE compared to existing baseline models [10]. Derong Xie et al established a dynamic multivariate partial grey model based on the parametric equations of traffic flow, the results reveal that the simulation error of the new model was less than 6 % [11]. These approaches have collectively laid a foundation for integrating deep learning into real-time traffic management, although challenges remain in achieving both high prediction accuracy and robustness under complex urban conditions.

Building upon these studies, this work proposes a KOA-CNN-BiGRU-MultiAttention hybrid neural network for short-term traffic flow prediction. The model leverages CNN to capture local spatial dependencies, BiGRU to learn bidirectional temporal features, and multi-head attention to identify key time steps for global temporal re-weighting while maintaining model efficiency. The Kepler optimization algorithm is specially employed to automatically search for optimal hyperparameters, thus enhancing generalization and stability across diverse traffic scenarios. Experimental results on the METR-LA data set demonstrate that the proposed approach achieves lower MAE compared with CNN baseline.

2. Methodology

2.1. The basic model of KOA-CNN-BiGRU-MultiAttention hybrid neural network

The proposed model integrates convolutional and recurrent architectures with attention and evolutionary optimization. A CNN module first extracts localized spatial-temporal features from input traffic sequences, which are later processed by a BiGRU module for capturing bidirectional temporal dependencies. A multi-head attention mechanism is applied to highlight critical time steps and enhance feature representation. Dropout and fully connected layers are employed for regularization and output mapping. The Kepler Optimization Algorithm automatically tunes hyperparameters to improve convergence and generalization. This hybrid framework enables robust and adaptive traffic prediction. The model structure is shown as Figure 1.

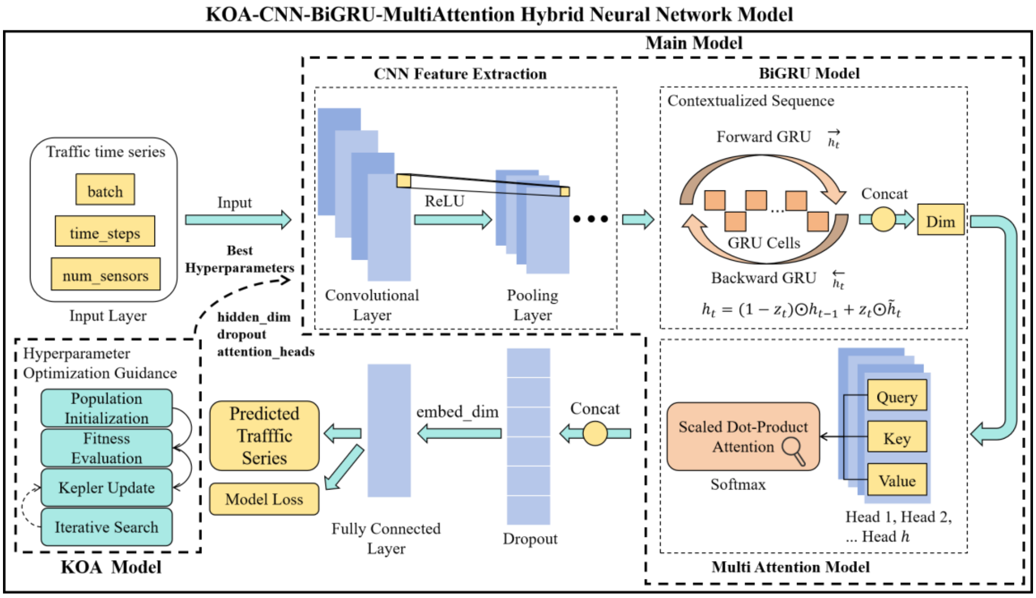


Figure 1. Neural network structure

2.2. Model structure and principle

This study optimizes the traditional CNN-GRU-Attention network by introducing the multiple attention mechanism for capturing dynamically weighting of important time steps, in which KOA is designed for automatically search for optimal hyperparameters for the model's higher accuracy and robustness, therefore achieving efficient extraction of data features and modeling of spatial-temporal relationships.

2.2.1 CNN for local spatial feature extraction

To extract local spatial features from the input traffic speed matrix, a 1D convolutional layer is employed along the temporal axis. Each input tensor of shape $(B \times N \times T)$ (where B is the batch size, N the number of sensors, and T the time steps) is first linearly embedded by transformation weights suggested by KOA, then reshaped to apply temporal convolution independently for each sensor. The convolution operation can be represented as (3). By applying a 1D convolution along the temporal axis (4), the output is then transposed back to shape $B \times N \times T$.

$$\mathrm{X_{KOA}} = \mathrm{X} \cdot \mathrm{W_{KOA}} + \mathrm{b_{KOA}},\ \mathrm{W_{KOA}} \in \mathbb{R}^{\mathrm{N\times N}} \tag{1}$$

$$\mathrm{X_{CNN-in}} = \mathrm{permute}(\mathrm{X_{KOA}}, (0,2,1)) \in \mathbb{R}^{\mathrm{B\times N\times T}} \tag{2}$$

$$\mathrm{X_{CNN}} = \mathrm{ReLU}(\mathrm{Conv1D}(\mathrm{X_{CNN-in}})) \in \mathbb{R}^{\mathrm{B\times N\times T}} \tag{3}$$

2.2.2 Bidirectional GRU for temporal feature learning

To capture both past and future dependencies in the time series, a bidirectional gated recurrent unit layer is used as (4), Dropout regularization is applied to prevent overfitting. This module processes the CNN output along the temporal dimension, learning sequential features in both directions. Its hidden state size H is also one of the hyperparameters optimized by KOA.

$$\mathrm{H_{GRU}} = \mathrm{BiGRU}(\mathrm{X_{CNN}}) \in \mathbb{R}^{\mathrm{B\times N\times 2H}} \tag{4}$$

2.2.3 Multi-head attention for global temporal re-weighting

To enhance the model's ability to focus on important time steps and capture global temporal relationships, we introduce a Multi-Head Attention mechanism. It computes multiple attention scores in parallel and aggregates them to form a contextual representation of each sequence.

Formally, for query, key, and value matrices Q, K, V derived from GRU outputs:

$$\mathrm{Attention(Q, K, V)} = \mathrm{Softmax}\left(\frac{\mathrm{QK^T}}{\sqrt{\mathrm{d_k}}}\right)\mathrm{V} \tag{5}$$

The final output is aggregated as (6).

$$\mathrm{MHA(X)} = \mathrm{Concat}(\mathrm{head_1}, \mathrm{head_2}, \ldots, \mathrm{head_a})\mathrm{W^O} \tag{6}$$

where h is the number of attention heads, and both h and projection matrix dimensions are hyperparameters searched by KOA.

2.2.4 Output layer with residual fusion

The model prediction is generated using the feature vector at the final time step of attention output, added with residual connection from the GRU output, followed by a fully connected output layer as (7). This residual fusion enhances gradient flow and prediction stability.

$$\mathrm{h_{fianl}} = \mathrm{H_{ATTN}}[:, -1, :] + \mathrm{H_{GRU}}[:, -1, :],\ \hat{\mathrm{y}} = \mathrm{h_{final}} \cdot \mathrm{W_{out}} + \mathrm{b_{out}} \tag{7}$$

2.3. Role of Kepler optimization algorithm

KOA is a physics-inspired metaheuristic algorithm raised by Mohamed et al in 2023. It optimizes the objective function by simulating the behavior of planets orbiting stars, so that candidate solutions, acting as "planets," gradually approach the optimal solution, acting as the "star." n this study, KOA is employed to automatically optimize critical hyperparameters of the proposed neural network, which includes learning rate α, CNN kernel size k, GRU hidden size h, number of attention heads a and dropout rate d.

$$P = \{\alpha, k, h, a, d\} \tag{8}$$

This joint optimization ensures a balanced trade-off between spatial–temporal representation capacity and generalization ability.

2.3.1 Initialization

Each candidate solution (“planet”) encodes a set of hyperparameters $\mathbf{P}_i$ randomly generated within predefined bounds. The position vector $\mathbf{X}_i$ corresponds to the $\mathbf{P}_i$ values, and the velocity vector $\mathbf{V}_i$ determines the update step size in the search space.

2.3.2 Fitness evaluation

For each planet, the proposed deep model is trained on the training set using its encoded hyperparameters. The fitness value f_i is computed on the validation set, with MAE as the primary objective:

$$f_i = \frac{1}{N}\sum_{j=1}^{N}(y_j - \hat{y}_j)^2 \tag{9}$$

Lower MAE values indicate higher fitness, corresponding to greater “mass” in the gravitational analogy.

2.3.3 Gravitational attraction

The best solution is treated as the "sun," and each planet updates its velocity and position based on gravitational forces defined as (10), where G is the gravitational constant (control parameter), m_i is the mass (fitness) of planet i, m^* is the mass of the sun, and $||X_i - X^*||$ is their Euclidean distance in the hyperparameter space.

$$F_{i*} = G \cdot \frac{m_i \cdot m^*}{||X_i - X^*||^2} \tag{10}$$

2.3.4 Velocity and position update

$$V_i^{t+1} = \omega V_i^t + \beta F_{i*} + \gamma \cdot \mathrm{rand}() \tag{11}$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \tag{12}$$

where ω represents inertia coefficient for balancing exploration and exploitation, β stands for gravitational influence scaling factor, and γ for random perturbation coefficient for diversity.

2.3.5 Local optima avoidance and elite preservation

To avoid local optima, KOA introduces random orbital perturbations and planet replacement when stagnation is detected. The elite preservation strategy ensures that the best-performing planet, which refers to the hyperparameter set, is retained and propagated across iterations.

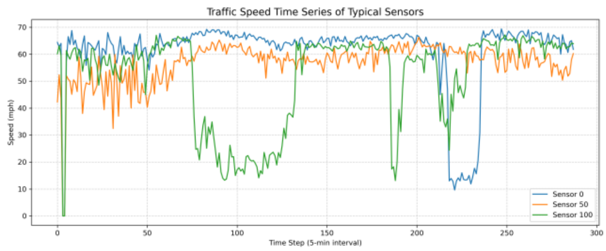
The KOA search process iterates until a convergence criterion is met. The final “sun” parameter set is then used to retrain the proposed model from scratch, ensuring optimal configuration for spatial–temporal traffic forecasting. This approach automates the traditionally manual hyperparameter tuning process, reduces trial-and-error cost, and improves model robustness under varying traffic scenarios.

3. Experiment results and analysis

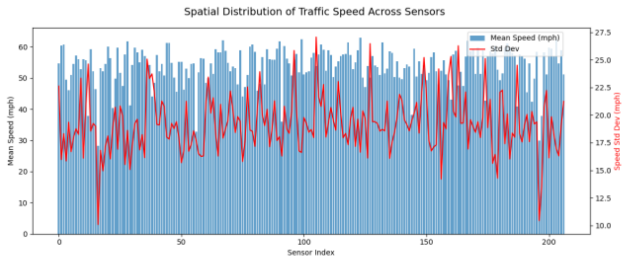
3.1. Data preprocessing and analysis

This study utilizes the open source METR-LA data set, which contains traffic speed records collected from 207 loop detectors at 5-minutes intervals on the highways of Los Angeles between March 1st and June 27th in 2017. Web link to this data set is https://github.com/liyaguang/DCRNN.

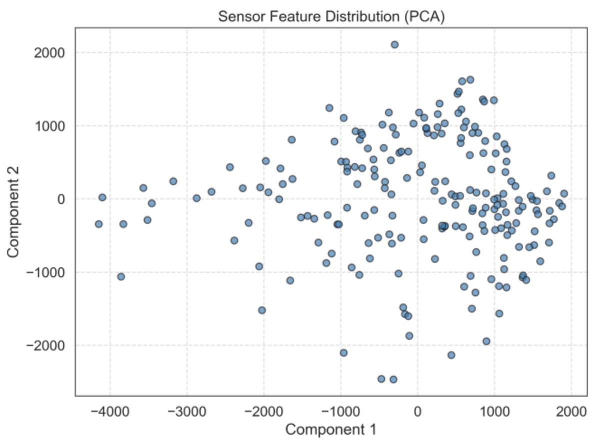
By choosing sensors that capture typical time-varying traffic patterns, peak hours, traffic incidents and road congestion can be reflected as shown in Figure 2. Temporal dynamics necessitate the use of sequence modeling techniques, which makes recurrent neural networks and attention mechanisms essential. To better understand the spatial-temporal structure of the data set, dimensionality reduction for sensor feature visualization is applied. Principal component analysis (PCA) linearly projects high-dimensional series into two dimensions, preserving global variance and highlighting overall traffic speed patterns. t-distributed stochastic neighbor embedding (t-SNE) emphasizes local neighborhood preservation, clustering sensors with similar dynamics closer together. The combination of PCA and t-SNE provides complementary perspectives: The former reveals global variance distribution, while the latter uncovers fine-grained local structures.



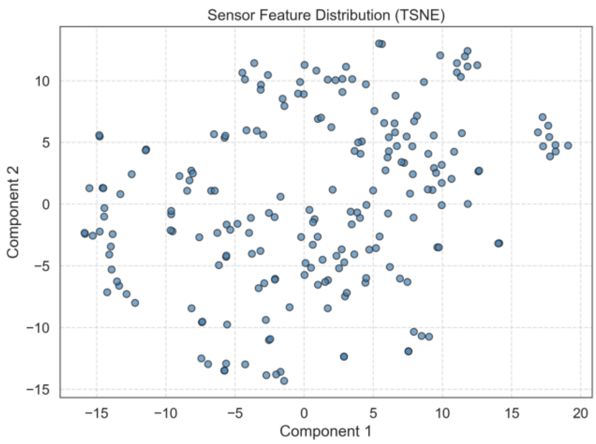
(a) Traffic speed time series of typical sensors



(b) Spatial distribution of total sensors



(c) PCA of sensor feature distribution



(d) t-SNE of sensor feature distribution

Figure 2. Traffic speed data feature visualization

To standardize the input space and accelerate model convergence, all data is scaled into the range [0, 1] using min-max normalization. By drawing the correlation heat map among total sensors, strong inter-sensor correlations indicating spatial dependencies can be observed as shown in **Figure 3**, thereby can be used to further illustrate the evolution of traffic flow. This insight justifies the integration of CNN and KOA modules in hybrid model to extract meaningful spatial features.

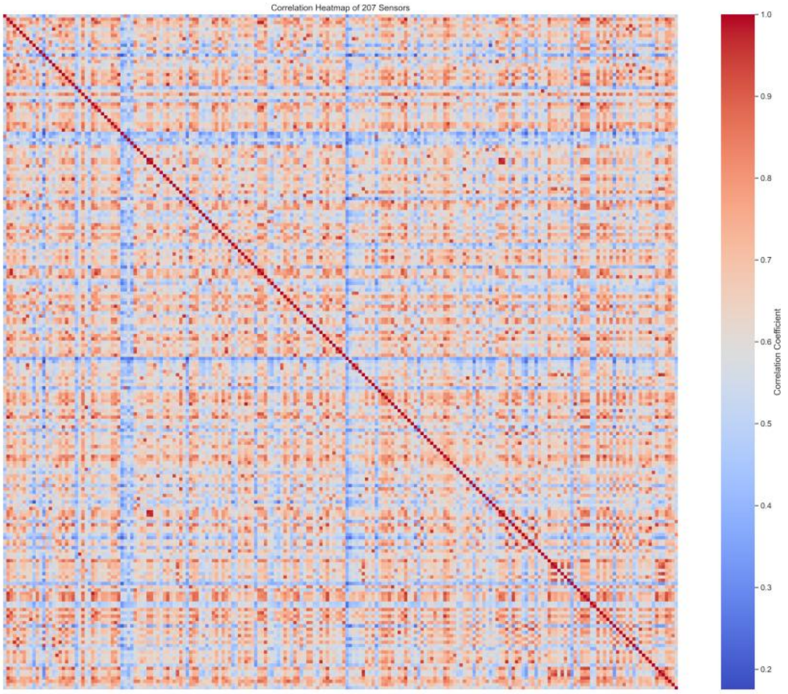


Figure 3. Correlation heat map of total 207 sensors

3.2. Simulation environment configuration

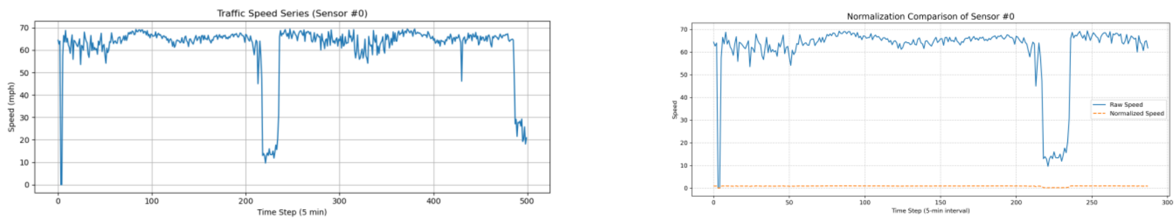
The experiment of this study is configured as shown in Table.1. The operating system applied in this study is Microsoft Windows10 22H2 (19045.5965) system, Pycharm 2025.1.1 version of the compiler.In order to apply pytorch framework, version of Python language used for the training is 3.10.9.

Table 1. The experimental environment configuration

Parameter Name	Parameter Value
Operating system	Microsoft Windows10 22H2 (19045.5965)
CPU	Intel Core i9-12900KF
GPU	NVIDIA GeForce RTX 2060
RAM	32 GB
Compilers	Pycharm2025.1.1 64-bit
Python version	3.10.9

3.3. Validation object selection and data normalization

As shown in Figure 4 (a), the raw traffic speed series from a representative sensor illustrates obvious temporal variations, which includes morning and evening peak patterns. These dynamics underscore the necessity of a temporal model that can capture non-stationary patterns in urban traffic. Figure 4 (b) compares the raw and normalized traffic speeds. Min-Max scaling is adopted to standardize the input to the range [0, 1], facilitating faster convergence and stability during model training.



(a) Traffic speed series of sensor #0 (b) Speed series after normalization

Figure 4. Validation object selection and data normalization

3.4. Train and validation performance

Mean absolute error (MAE) and root mean square error (RMSE) are used as metrics to evaluate model performance. MAE is especially employed as the model loss function, as it directly measures the average magnitude of prediction errors without considering their direction, which is shown as (13).

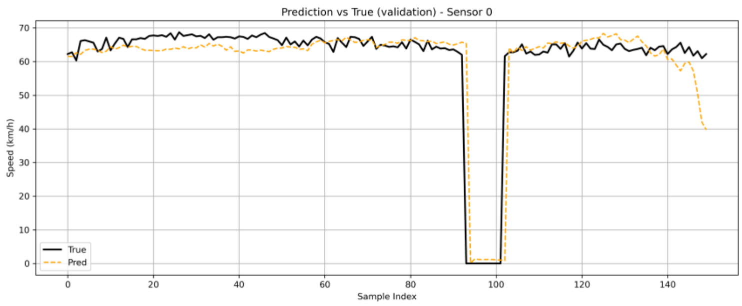
$$\mathcal{L}_{MAE} = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{13}$$

The training loss and validation loss over 50 epochs are presented in Figure 5. The training curve demonstrates smooth and consistent convergence, while the validation MAE and RMSE decrease stably in early epochs before plateauing after approximately 15 epochs. The use of early stopping prevents overfitting and reduces unnecessary computation. The relatively low validation metrics confirm that the hybrid model generalizes well to irregular traffic sequences.

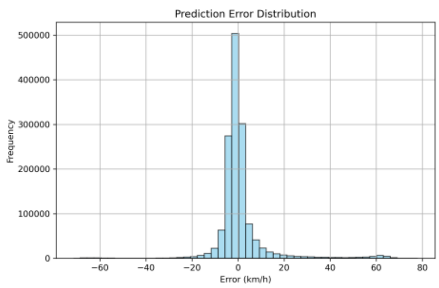
Figure 5. Model training performance

3.5. Prediction accuracy on validation set

To further illustrate the model's prediction performance, the predicted and true traffic speed values over 100 consecutive time steps was compared as shown below. Figure 6 (a) shows that the prediction line closely follows the ground truth, including during rapid rises and falls in speed, indicating that the model can effectively capture traffic dynamics even under fluctuating conditions. In Figure 6 (b), the prediction error distribution demonstrates a near-normal shape centered around zero, with most errors within ±5 km/h. This indicates stable model performance without systematic bias, while few long-tail deviations suggest challenges in capturing sudden traffic fluctuations such as accidents or unexpected congestion. These results demonstrate that the combined use of bidirectional GRU and multiple attention mechanism enables the model to respond to both short-term shifts and global temporal context, while CNN helps model spatial continuity.



(a) Comparison between prediction and truth



(b) Error distribution of prediction

Figure 6. Prediction accuracy

3.6. Model performance comparison

3.6.1 Parameter settings of KOA

This study uses KOA algorithm to optimize the hyperparameters of CNN-BiGRU-MultiAttention model, which include learning rate, number of convolutional kernels and number of BiGRU hidden layer nodes. The MAE of the validation set is used as the KOA fitness function to simulate the motion patterns of the sun and planets, dynamically adjusting the population positions to find the optimal solution.

The KOA algorithm is initialized with a population size of 10 and an iteration count of 50. The initial values for learning rate, number of convolutional kernels, number of hidden layer nodes, eccentricity, and orbital period are shown in Table 2. The eccentricity in the KOA algorithm controls the shape of the orbit, influencing the search path of planetary motion; the orbital period controls the speed of planetary motion, affecting the frequency and range of exploration.

Table 2. Initialization of population, eccentricity and orbital period in KOA

Initial Learning Rate	Number of Initial Convolution Kernels	Number of Hidden Layer Nodes	Initial Eccentricity	Initial Orbital Period
0.408	9	96	0.637	0.075
0.287	32	40	0.362	0.506
0.526	42	96	0.015	1.052
0.686	63	80	0.542	0.971
0.814	47	8	0.318	0.077
0.904	32	16	0.145	0.435
0.835	18	24	0.549	0.553
0.449	45	120	0.187	0.267
0.833	35	80	0.399	0.009
0.835	32	96	0.240	0.641

A smaller TC enhances the ability of global search, while a larger TC is beneficial for reinforcing local search. M is the gravitational strength control factor, influencing the adjustment of gravitational strength, individual position updates, and search range control. As M decays, the search range of individuals gradually narrows, and the algorithm transitions from global search to local search. M_0 and λ control the decay rate of M, thereby influencing the balance of the search process. Specifically, TC controls the periodic changes in gravity, thereby affecting the update of planetary positions and exploration speed; M_0 controls the gravitational strength between planets; λ determines the decay rate of gravity, affecting the exploration capability of planets during the search process. Therefore, the above control parameters are critical in the KOA algorithm that influence the search process and convergence speed. To further analyze the specific impact of these control parameters on the optimization performance of the KOA algorithm, Table 3 below shows the fitness results of the KOA algorithm under several different combinations of control parameters.

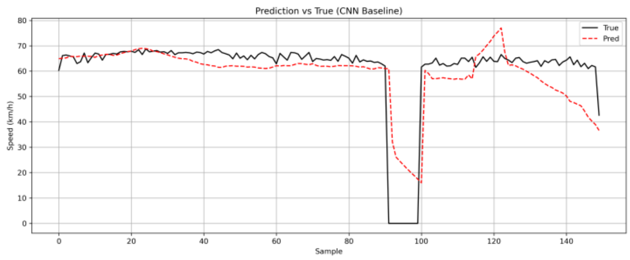
Table 3. Impact of different control parameter combinations on fitness in KOA algorithm

Combination	TC	M_0	λ	Best Fitness
Combination_01	0.1	0.05	1	0.0659
Combination_02	1	0.05	8	0.0621
Combination_03	3	0.1	15	0.0631
Combination_04	0.1	10	100	0.1085
Combination_05	3	10	100	0.0750
Combination_06	10	0.1	100	0.0689

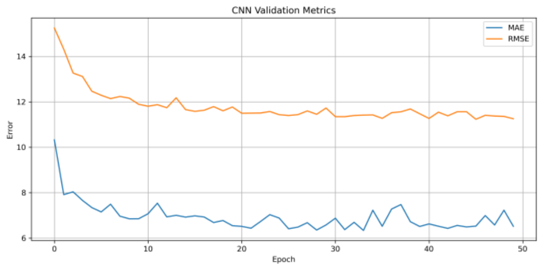
Table 3 shows six different combinations of control parameters and their corresponding optimal fitness values. As shown in the table, smaller TC values such as 0.1, 1 and 3, accelerate the convergence speed of the planet and enhance the global search capability, but may lead to premature convergence to local optima in some cases. In contrast, larger TC value like 10, maintains a broader exploration range for the planets, helping to avoid premature convergence, but may result in reduced search efficiency. Larger M_0 values like 10 enhances the gravitational forces between planets and increases interactions, which may lead to premature convergence and confinement to certain solution neighborhoods. Smaller M_0 values like 0.05 and 0.1 weaken gravitational forces, increasing planetary freedom and maintaining a broader search range. Larger λ value like 100 accelerates gravitational decay, causing planets to converge faster and risk getting stuck in local optima, while smaller λ values such as 1, 8 and 15, slow down gravitational decay, helping to enhance search persistence and balance the capabilities of global and local search. Through comprehensive analysis, the optimal KOA control parameter combination in this model is Control Parameter Combination_02, i.e., TC=1, M_0=0.05, λ=8.

3.6.2 Comparison with baseline model

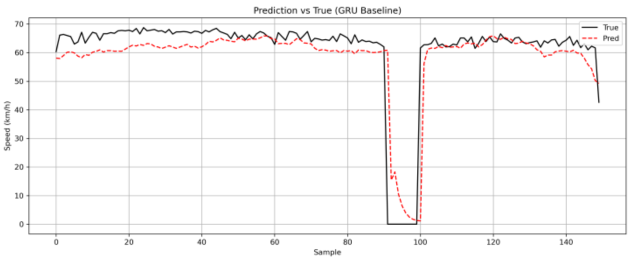
To evaluate the effectiveness of each component in the proposed hybrid architecture, comparison of the hybrid model against simplified baseline models that use single CNN, single GRU and CNN-BiGRU-Attention for spatial-temporal modeling are designed. These baseline models share the same input-output structure and are trained under identical hyperparameters (window size, batch size, learning rate, epochs) to ensure fairness of, relative training results are shown in Figure 7.



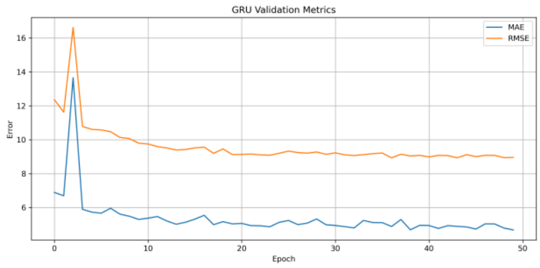
(a) Prediction accuracy of CNN



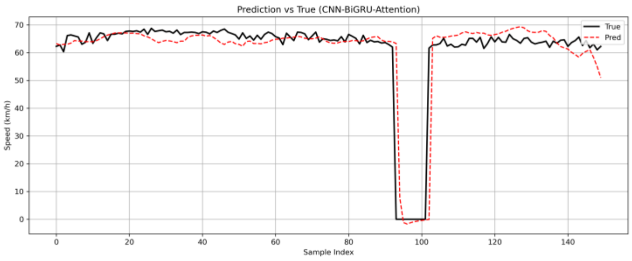
(b) Validation metrics of CNN



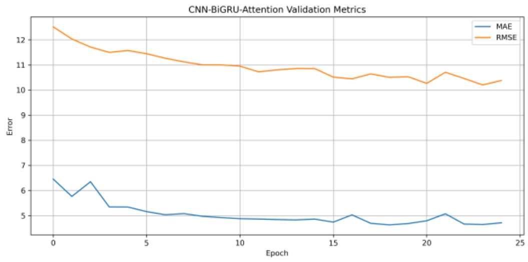
(c) Prediction accuracy of GRU



(d) Validation metrics of GRU



(e) Prediction accuracy of CNN-BiGRU-Attention



(f) Validation metrics of CNN-BiGRU-Attention

Figure 7. Metrics comparison between baseline models

The results of both models on the METR-LA validation set are presented in Table 4. The evaluation metrics of MAE and RMSE indicate a significant fluctuation compared with the hybrid model raised in this study, proving its higher accuracy in predicting short term traffic flow.The results in Table 4 demonstrate that the proposed model achieves the lowest MAE (4.8790) and RMSE (8.8356), outperforming all baselines. Compared with the CNN-BiGRU-Attention model, our approach reduces MAE by 7.8% and RMSE by 10.8%, highlighting the contribution of KOA-based hyperparameter optimization and the multi-head attention mechanism.

In terms of computational efficiency, the baseline CNN-BiGRU-Attention requires 5.3M parameters, while our optimized hybrid reduces this to 4.7M through KOA-driven search, yielding a more compact architecture with improved accuracy. This confirms that the proposed model does not merely improve prediction precision, but also enhances parameter efficiency.

Table 4. Training metrics of proposed and baseline model

Model Component	MAE	RMSE	Train Loss
Single CNN	6.5202	11.3245	0.1892
Single GRU	5.3782	9.0673	0.1207
CNN-BiGRU-Attention	5.2908	9.9107	0.1039
KOA-CNN-BiGRU-MultiAttention	4.8790	8.8356	0.0524

4. Conclusions

This study proposes a structurally optimized hybrid neural network for short-term traffic flow prediction, specifically targeting the complex and dynamic nature of urban traffic flow speed. By integrating spatial feature extraction via CNN, temporal sequence modeling through bidirectional GRU, and dynamic temporal weighting using multi-head attention, the model is capable of accurately capturing spatial-temporal dependencies in multi-scale. Furthermore, the introduction of Kepler optimization algorithm enables automatic configuration of key model parameters, enhancing both predictive performance and training efficiency. Multiple experiments on the METR-LA data set demonstrate that the proposed model achieves significantly lower MAE compared to a CNN-BiGRU-Attention baseline model, validating the former’s generalizability and robustness across sensor nodes and time horizons.

Beyond numerical accuracy, the study emphasizes interpretability by incorporating visualizations, which includes sensor-level speed trends, spatial variability and inter-sensor correlations, therefore offering unique insights into the evolution of traffic flow. Future work will focus on extending model settings to multiple tasks and procedures, integrating external factors like weather or events, and exploring intercity transfer learning to enhance deployment scalability in real-world intelligent transportation systems.

References

[1] WANG Xiang-xue, XU Lun-hui. Short-term Traffic Flow Prediction Based on Deep Learning[J]. Journal of Transportation Systems Engineering and Information Technology, 2018,18(01):81-88.

[2] FENG Ning, GUO Sheng-Nan, SONG Chao, et al. Multi-component Spatial-temporal Graph Convolution Networks for Traffic Flow Forecasting[J]. Journal of Software, 2019, 30(03):759-769.

[3] YUAN Hua, CHEN Zehao. Short-term Traffic Flow Prediction Based on Temporal Convolutional Networks[J]. Journal of South China University of Technology (Natural Science Edition), 2020, 48(11):107-113+122.

[4] Nisha Singh, Kranti Kumar, Bhawna Pokhriyal. Attention based spatiotemporal model for short-term traffic flow prediction[J]. International Journal of System Assurance Engineering and Management, 2025, 16(4):1-15.

[5] Sonia Mrad, Rafaa Mraihi, Aparna S. Murthy. Efficient implementation of a wavelet neural network model for short-term traffic flow prediction: Sensitivity analysis[J]. International Journal of Transportation Science and Technology, 2025, 17:21-38.

[6] Naheliya Bharti, Redhu Poonam, Kumar Kranti. Deep Bi-LSTM Neural Network with Chaotic Particle Swarm Optimization Technique for Short-Term Traffic Speed Prediction. Springer Nature Singapore, 2025, :171-191.

[7] Guowen Dai, Jinjun Tang. A Short-Term Traffic Flow Prediction Method Based on Personalized Lightweight Federated Learning[J]. Sensors, 2025, 25(3):967-967.

[8] Xiaoqing Wang, Feng Sun, Xiaolong Ma, et al. Short-term traffic flow prediction based on vehicle trip chain features[J]. Transportation Letters, 2025, 17(1):157-168.

[9] Ali Reza Sattarzadeh, Ronny J. Kutadinata, Pubudu N. Pathirana, et al. A novel hybrid deep learning model with ARIMA Conv-LSTM networks and shuffle attention layer for short-term traffic flow prediction[J]. Transportmetrica A: Transport Science, 2025, 21(1):388-410.

[10] Lingjuan Chen, Cong Xie, Dongfang Ma, et al. A short-term traffic flow prediction model for road networks using inverse isochrones to determine dynamic spatiotemporal correlation ranges[J]. Physica A: Statistical Mechanics and its Applications, 2025, 657:130244-130244.

[11] Derong Xie, Hongli Chen, Huiming Duan. A dynamic multivariate partial grey model based on the traffic flow parameter equation and its application[J]. Physica A: Statistical Mechanics and its Applications, 2024, 656:130204-130204.