Proceedings of the 2025 SICE Festival with Annual Conference
September 9-12, 2025, Chiang Mai, Thailand

Improving Vehicle Trajectory Prediction with EMA-Attention Mechanism and LSTM Networks

Jianping Yao[1†] and Katsumi Konishi[2]

[1]Department of Computer and Information Sciences, University of Hosei, Tokyo, Japan
(Tel: +81-70-9116-3143; E-mail: jianping.yao.4u@stu.hosei.ac.jp)
[2]Department of Computer and Information Sciences, Hosei University, Tokyo, Japan
(Tel: +81-42-387-4532; E-mail: konishi@hosei.ac.jp)

Abstract: Accurate trajectory forecasting is critical for the safety and efficiency of autonomous vehicles. Existing models, though accurate, struggle with noise sensitivity, interpretability, and structural complexity, limiting their real-world applicability. This study introduces an LSTM model enhanced with an EMA attention mechanism to address these challenges. The model effectively captures long- and short-term dependencies in vehicle trajectories, emphasizing key temporal features and dynamically adjusting their weights through EMA attention. Unlike traditional mechanisms, EMA reduces computational complexity, mitigates trajectory noise, highlights critical time-steps, and improves model robustness and interpretability. Evaluated on the NGSIM dataset, the EMA-Attention LSTM model demonstrates superior predictive accuracy with low computational overhead, making it suitable for real-time traffic predictions in complex driving environments.

Keywords: Vehicle trajectory prediction, EMA attention, long short-term memory, intelligent transportation systems

1. INTRODUCTION

As autonomous driving technology zooms ahead at breakneck speed, the ability to accurately predict vehicle paths has become a cornerstone for guaranteeing both the safety and seamless operation of smart transportation networks and self-driving cars [1]. Autonomous driving systems enhance safety and reliability by precisely forecasting the movements of surrounding vehicles, enabling dynamic route adjustments and collision avoidance. Real-time trajectory prediction is especially vital in complex settings like highways, where dense traffic and intricate interactions demand efficient, robust, and interpretable predictive models for secure navigation.

Recently, trajectory prediction tasks have widely utilized advanced deep learning techniques. Long Short-Term Memory Networks (LSTMs), a subset of Recurrent Neural Networks (RNNs), have become the go-to methods for processing vehicle trajectory data. Their robust capacity to capture sequential patterns has propelled them to the forefront of the field [2]. In addition, the introduction of advanced models such as Transformer has further improved the performance of trajectory prediction, especially in capturing long time dependencies [3]. However, the existing methods still face many limitations in practical applications.

First of all, driving behaviors are inherently stochastic and multimodal in nature, such as harsh braking, sudden acceleration, or lane-changing overtaking behaviors, which are usually regarded as noise in vehicle trajectory data and can have a great impact on the feature extraction and prediction result of the model [4]. Existing methods are either not robust enough to noise to cope with complex traffic scenarios, or rely on high computational complexity attention mechanisms (e.g., spatio-temporal attention, global attention) to enhance the ability to model global relationships among input features, leading to severe limitations in real-time.

Deep learning models like LSTM and global attention mechanisms often suffer from limited interpretability, hindering the identification of critical time-steps or trajectory features that shape predictions. This opacity undermines understanding of decision-making processes, reducing their reliability and applicability. To address these issues, this study proposes an improved LSTM network integrated with an Exponential Moving Average (EMA) attention mechanism, designed to enhance vehicle trajectory prediction in complex highway scenarios. The model leverages LSTM to capture both long- and short-term dependencies in the motion patterns of the target vehicle and its surroundings.

It adeptly modifies the significance of each time-step by merging the EMA attention technique, thus emphasizing the critical points in the temporal sequence, and at the same time smoothes out the noise interference, which effectively improves the robustness to the anomalous trajectories. Moreover, the EMA attention mechanism intuitively distributes temporal feature weights and ensures computational efficiency, thereby greatly enhancing the model's interpretability. Compared with the traditional attention mechanisms (e.g., global attention and spatio-temporal attention), the method proposed in this paper demonstrates clear advantages in computational efficiency, noise resilience, and interpretability, providing a reliable approach for real-time trajectory forecasting tasks.The primary contributions of this paper are as follows:

- This research introduces a forecasting method that merges the EMA attention model with LSTM networks, enhancing LSTM's dependency capturing by dynamically weighting temporal features and reducing noise interference.

† Jianping Yao is the presenter of this paper.

- The EMA attention mechanism uses linear computation, eliminating the need for global relationship calculations at all time steps, which greatly reduces computational complexity and enhances real-time prediction capabilities.
- The EMA attention mechanism dynamically adjusts temporal feature weights, preserving high-precision predictions amidst noisy trajectory data, thereby bolstering model robustness.
- The EMA attention mechanism produces a clear and intuitive time step weight distribution, highlighting key time steps' contributions to predictions and significantly improving model interpretability.
- The model tested on the NGSIM dataset, outperforms current methods in multi-step time predictions, showing high efficiency and robustness for real-world traffic applications.

2. RELATED WORK

Vehicle trajectory prediction is an important part of autonomous driving, and researchers have explored a number of different approaches in order to achieve high-precision prediction. At the earliest, researchers used traditional methods based on physical models, statistical models, and trajectory fitting to predict vehicle trajectories [5]. Physical models (e.g., constant velocity models) assume that the vehicle follows a simple law of motion and predict the future position of the vehicle from information such as velocity, acceleration, and position [6]. Although this method is simple in terms of computation, the driving behavior is multimodal (e.g., harsh braking, sudden acceleration) with the assumption that the premise is too idealized, ignoring the interaction between vehicles is difficult to deal with complex driving behavior. With the development of sensor technology, statistical models (e.g., Kalman filtering) have been introduced into the field of trajectory prediction. Kalman filtering realizes trajectory prediction through state estimation and recursive prediction, and combines sensor data [7]. However, its performance capability is very limited if the vehicle performs nonlinear behaviors such as sharp turns or acceleration. Trajectory fitting methods (e.g., polynomial regression and Bessel curves) generate future trajectories by fitting historical trajectories, which are suitable for short-term prediction of smoothed trajectories, but have insufficient ability to model abrupt trajectory change behavior and long-term dependence.

With the development of technology, deep learning techniques have gradually come into the limelight. Recurrent neural networks and long short-term memory models, among other time series techniques, are now the prevalent choice for forecasting vehicle paths, leveraging their superior ability to model temporal relationships [8]. Altché and de La Fortelle, among others, introduced an LSTM-based approach to predict highway vehicle paths, showcasing its proficiency in mid-range trajectory forecasting with exceptional precision [9]. Guo XIE and colleagues introduced a model combining a CNN and LSTM network sequentially, which extracts spatial features by CNN, processes time series data by LSTM, and optimizes hyper-parameters by using lattice search algorithms for predicting trajectories around self-driving vehicles [10]. However, traditional time series models (e.g., RNN or LSTM) have fixed weight assignments for time steps, which makes it difficult to dynamically focus on critical time steps. In addition, they are more sensitive to noise, resulting in insufficient robustness in complex scenarios.

Time series models perform well in handling typical time series data such as historical vehicle trajectories. However, due to the limitations of these models with respect to the fixity of time-step weight assignment, researchers have gradually tried to introduce an attention mechanism to dynamically adjust the time-step weights to compensate for this shortcoming.Leilin et al. proposed a long short-term memory (LSTM) network integrated with a spatiotemporal attention mechanism, which solves the problem of the model's lack of interpretability in the prediction of vehicle trajectories by introducing a spatiotemporal attention mechanism and at the same time achieving high accuracy prediction and interpretable analysis of key influencing factors [11]. Messaoud et al. introduced an LSTM-based model enhanced with a multi-head attention mechanism [12]. This approach addresses the challenge of insufficient dynamic dependency modeling for surrounding vehicles in vehicle trajectory prediction by explicitly capturing long-distance spatio-temporal interactions between vehicles. It accurately simulates the intricate dynamics between the target vehicle and nearby traffic, enhancing forecast precision.The attention mechanism boosts the model's ability to select features effectively, spatio-temporal attention mechanisms or multi-head attention mechanisms typically require computing the global relationships across all time steps or features. This leads to significant computational demands, posing difficulties for real-time prediction requirements.

This study proposes a modified LSTM model, incorporating an EMA attention technique, to tackle the presented obstacles. Compared with traditional attention mechanisms, the EMA attention mechanism significantly reduces the computational complexity by dynamically weighting to highlight the information of key time steps while smoothing the noise. Furthermore, the technique excels in producing easily comprehensible time-step weight distributions, maintaining linear complexity. This advancement not only boosts the precision and resilience of the forecasts but also enhances the clarity of the model. The efficacy of the approach has been confirmed through rigorous testing on the NGSIM dataset, showcasing its promising utility for real-time vehicle trajectory forecasting.

3. METHODOLOGY

The model consists of five core modules: encoder, EMA module, attention mechanism, social interaction encoder, and decoder. The configuration is depicted in Fig. 1, featuring a two-layer LSTM architecture for analyzing the target vehicle's historical path and the movement data of surrounding vehicles, encode them into high-dimensional time-series features, capture the long- and short-term dependencies, and provide dynamic feature support for the subsequent modules. The EMA module highlights the information of the key time steps through the group-normalized exponential moving average attention mechanism while smoothing out the noise to enhance the robustness of the features and the spatial expressiveness. The attention mechanism also allocates weights to the encoder's output, emphasizing critical time points and enhancing the model's flexibility in handling complex traffic situations. The social interaction coding filters and fuses the trajectory features of neighboring vehicles through a masking mechanism to model the collaborative behaviors and dynamic interactions among vehicles. Finally, the decoder synthesizes the information from the encoder, EMA module, and social interaction coding to generate highly accurate predictions of future trajectories using LSTM.

3.1. LSTM Encoder

We utilize the past movement patterns of a vehicle, denoted as $\mathbf{H_t} \in \mathbb{R}^{\mathrm{T}\times 2}$, along with the motion trajectories of adjacent vehicles, denoted as $\mathbf{N_t} \in \mathbb{R}^{\mathrm{T}\times\mathrm{M}\times 2}$, as input data. These input data are processed using a two-layer LSTM structure to extract temporal features. Initially, they are projected into a high-dimensional space via a feature embedding function:

$$\mathbf{e_t} = \phi\left(\mathbf{h_t}\right), \quad \mathbf{e_t^n} = \phi\left(\mathbf{n_t}\right), \tag{1}$$

where $\phi(\cdot)$ represents the embedding function that transforms trajectory data into a high-dimensional representation. The embedding vectors are then fed into a dual-layer LSTM to individually capture the motion characteristics of both the target and surrounding vehicles:

$$\mathbf{z_t} = \mathrm{LSTM}_\mathbf{1}\left(\mathbf{e_t}\right), \quad \mathbf{z_t^n} = \mathrm{LSTM}_\mathbf{2}\left(\mathbf{e_t^n}\right). \tag{2}$$

The first LSTM layer is responsible for analyzing the historical movement of the target vehicles to infer their motion tendencies, whereas the second LSTM layer focuses on the interactive behaviors of surrounding vehicles, thereby capturing their social dynamics. This two-layer structure enables efficient processing of long-sequence data while preserving critical information from historical trajectories.

Finally, the encoder produces a sequence of temporal feature representations:

$$\mathbf{Z} = \left\{\mathbf{z_1}, \mathbf{z_2}, \ldots, \mathbf{z_T}\right\}, \quad \mathbf{Z^n} = \left\{\mathbf{z_1^n}, \mathbf{z_2^n}, \ldots, \mathbf{z_T^n}\right\}. \tag{3}$$

At each time step t, the encoder captures the evolving characteristics of the target and neighboring vehicles by updating the hidden state $\mathbf{h_t}$ and the memory cell $\mathbf{c_t}$. These high-dimensional representations not only effectively retain the essential temporal information but also provide subsequent modules with crucial insights into the motion dynamics and social interactions of the target vehicle.

3.2. EMA Module and Attention Mechanism

The features of the encoder output are enhanced for spatial expressivity and temporal dynamic weight assignment through the EMA module and the attention mechanism, respectively.

First, the EMA module is used to dynamically adjust the weights of key time steps and spatial locations while smoothing out noise disturbances using the attention mechanism of group normalization and exponential moving average. For feature $\mathbf{X}$, the EMA component initially calculates the dynamic weights $\mathbf{w_{ij}}$ as per this equation:

$$\mathbf{w_{ij}} = \sigma\left(\mathrm{GN}\left(\mathbf{x_{ij}}\right) \cdot \mathrm{Conv}_{\mathbf{1}\times\mathbf{1}}\left(\mathbf{x_{ij}}\right)\right), \tag{4}$$

where $\sigma(\cdot)$ represents the Sigmoid activation, $\mathrm{GN}(\cdot)$ stands for Group Normalization, and $\mathrm{Conv}_{1\times 1}(\cdot)$ denotes the 1×1 convolution operation. Second, the input features are dynamically weighted according to the weight distribution, and the output features are represented as:

$$\mathbf{X'} = \mathbf{X} \cdot \mathbf{W}. \tag{5}$$

The process highlights information at key locations while effectively reducing the impact of noise.

Second, the attention mechanism dynamically weights the time series features $\mathbf{Z}$ output by the encoder to focus on critical time steps. The attention weights are computed using the Softmax function:

$$\alpha_t = \frac{\exp\left(w_t\right)}{\sum_{i=1}^{T} \exp\left(w_i\right)}, \tag{6}$$

where $w_t = f(z_t)$ and $f(\cdot)$ denotes the feature mapping function. The time series features are weighted and summed using the attention weights to generate a new feature representation $\mathbf{h}$, the formulas are listed below:

$$\mathbf{h} = \sum_{\mathbf{t}=\mathbf{1}}^{\mathbf{T}} \alpha_\mathbf{t} \cdot \mathbf{z_t}. \tag{7}$$

The output $\mathbf{h}$ is the attention-weighted target vehicle time series features.

3.3. Social Interaction Coding

To enhance prediction precision, considering the impact of adjacent vehicles on the target vehicle is crucial [13]. To analyze the interplay between the target vehicle and its surroundings, we employ a social interaction coding module. This tool leverages a masking mechanism

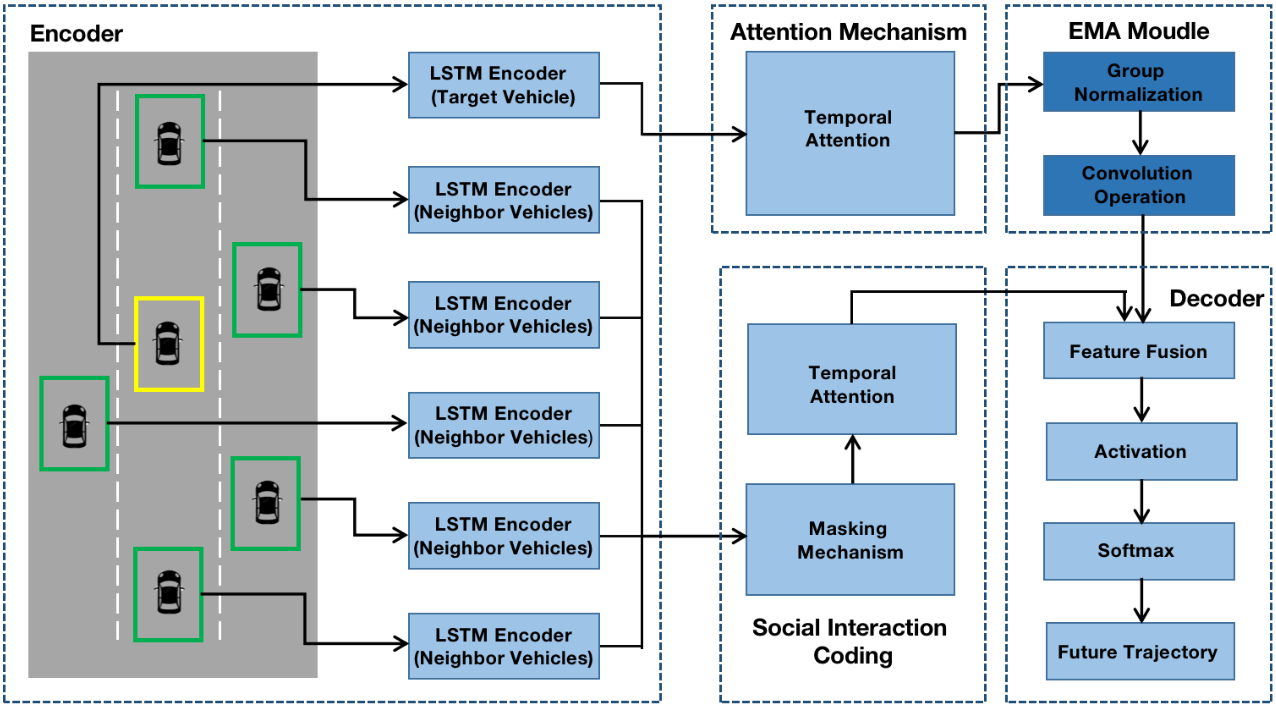

Fig. 1: The structure of EMA-LSTM: The model takes the past movement paths of the target vehicle (indicated by the yellow box) and the surrounding vehicles (indicated by the green box) as inputs. Initially, a dual-layer LSTM encoder is utilized to capture the temporal characteristics of the target vehicle and nearby vehicles. Subsequently, the two temporal characteristics are given dynamic weight adjustments through the attention mechanism, while the EMA module and the social interaction encoding unit refine their features separately. Ultimately, the decoder integrates the dynamic attributes of the focal vehicle with the interactive cues of the neighboring vehicles to forecast its prospective path.

to sift through the trajectory data of nearby vehicles, allowing us to effectively capture and interpret the nuanced social dynamics at play. Through screening, fusion and feature enhancement, this module provides accurate social context information to the decoder.

For the trajectory features $\mathbf{Z_n}$ of neighboring vehicles, the invalid or irrelevant vehicles are filtered by the mask matrix $\mathbf{M}$. The filtered neighboring vehicle features are represented as:

$$\mathbf{S} = \mathbf{M} \odot \mathbf{Z_n}, \tag{8}$$

where $\odot$ denotes per-element multiplication. The value of $\mathbf{M}$ is generated based on the relative position of neighboring vehicles or a distance threshold to ensure that only trajectory features that have a direct impact on the target vehicle are retained.

Subsequently, the social interaction features $\mathbf{S}$ are spliced with the dynamic features of the target vehicle $\mathbf{h} \in \mathbb{R}^{\mathrm{T}\times\mathrm{d}}$ to generate the composite features $\mathbf{F}$, which are used to further model the dynamic behaviors of the target vehicle in the social environment:

$$\mathbf{F} = \mathrm{concat}\left(\mathbf{S}, \mathbf{h}\right), \tag{9}$$

where $\mathrm{concat}(\cdot)$ denotes the feature splicing operation, which fuses the target vehicle features with the neighboring vehicle features.

This module adeptly pinpoints the collaborative nuances and interplay among the focal vehicle and its surrounding peers, zeroing in on the intricate dynamics of group vehicle interactions as represented in the NGSIM dataset. This design is crucial for modeling the characteristics of social dynamics in real traffic scenarios and enhancing the precision of trajectory prediction.

3.4. Decoder

The decoder module generates trajectory prediction results for multiple future time steps based on the composite feature $\mathbf{F}$ using a single layer LSTM. The primary function of the decoder is to take the dynamic attributes of both the target and nearby vehicles, which have been gleaned by the encoder and the social interaction module, and convert them into precise forecasts of their upcoming paths.

The decoder first receives the synthesized features $\mathbf{F} \in \mathbb{R}^{\mathrm{d}}$ and feeds them into a single-layer LSTM for time-series processing to generate the hidden states $\mathbf{o_t}$:

$$\mathbf{o_t} = \mathrm{LSTM}\left(\mathbf{F}\right), \tag{10}$$

where $\mathbf{o_t} \in \mathbb{R}^{\mathrm{h}}$ denotes the output of the LSTM and h is the dimension of the hidden layer.

Second, the hidden state $\mathbf{o_t}$ output by LSTM is converted to the trajectory prediction result $\hat{\mathbf{y}}_t$ through the linear mapping layer as follows:

$$\hat{\mathbf{y}}_t = \mathbf{W_o}\mathbf{o_t} + \mathbf{b_o}, \tag{11}$$

where, $\mathbf{W_o} \in \mathbb{R}^{2\times \mathrm{h}}$ represents the weight matrix, and $\mathbf{b_o} \in \mathbb{R}^2$ denotes the bias vector of the linear layer. These parameters are responsible for transforming the hidden state $\mathbf{o_t} \in \mathbb{R}^{\mathrm{h}}$ into the output space for trajectory prediction.The output $\hat{\mathbf{y}}_t \in \mathbb{R}^2$ corresponds to the predicted position of the target vehicle at time step t, comprising its x and y coordinates.

The decoder generates the target vehicle's predicted trajectory for the subsequent T' time steps as follows:

$$\hat{\mathbf{Y}} = \{\hat{\mathbf{y}}_{T+1}, \hat{\mathbf{y}}_{T+2}, \ldots, \hat{\mathbf{y}}_{T+T'}\}. \tag{12}$$

In this context, T signifies the total count of time intervals within the historical path, while T' indicates the number of intervals allocated for forecasting. The matrix $\hat{\mathbf{Y}}$, which belongs to the set $\mathbb{R}^{\mathbb{T}'\times 2}$, represents the projected course of the target vehicle in the future. Each row within this matrix comprises the x and y positional coordinates pertaining to a distinct time slice.

4. EXPERIMENTS

4.1. Data Introduction

NGSIM is a high-quality publicly available dataset widely used for traffic flow modeling and vehicle behavior analysis, and is designed with the goal of supporting research and development of transportation systems with highly accurate data [14]. It includes all vehicles operating on US-101, I-80, and additional roadways in a time period of vehicle travel conditions. The data is acquired using a camera, and the dataset records information about the dynamic movement of vehicles on the highway.

4.2. Parameterization and Evaluation

In this study, the encoder's hidden layer is configured with 64 dimensions, while the decoder's hidden layer is set to 128 dimensions. The input history spans 16 time steps, and the output prediction extends over 5 time steps. The input history trajectories are mapped to a 32 dimensional feature space through the embedding layer. The social interaction grid is set up as a 13×3 matrix, allowing the model to adeptly understand the shifting relationships between nearby vehicles and the target vehicle. For optimization, the Adam optimizer is utilized with a learning rate of 0.001. To handle incomplete output sequences during training, the model employs the Masked Mean Squared Error (Masked MSE) as its loss function. The training regimen spans 10 epochs, with each epoch encompassing both training and validation stages. All experiments are based on the PyTorch framework and Python 3.8 environment, and the models are trained and tested on NVIDIA RTX 4070 Ti GPU.

This investigation employs the Root Mean Square Error (RMSE) as the key performance measure for assessing the discrepancy between the forecasted path and the actual trajectory. RMSE, measured in meters, is computed using the following formula:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}. \tag{13}$$

Throughout the training phase, distinct RMSE patterns for both the training and validation datasets are documented to assess the model's convergence and its capacity to generalize.

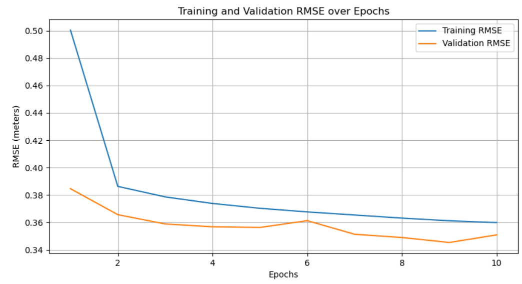

Fig. 2: Training and validation RMSE over 10 epochs for EMA-LSTM.

To further evaluate the training performance and prediction stability of the EMA-LSTM model, we visualized both the RMSE convergence during training and the prediction accuracy across future time steps. As shown in Fig. 2, the RMSE curves for both training and validation datasets over 10 epochs demonstrate a smooth and stable convergence trend, suggesting effective learning without overfitting. This confirms the model's ability to generalize well to unseen data during trajectory forecasting.

4.3. Case Study on LSTM Failure Samples

The test set used in this study contains a total of 1,505,756 vehicle trajectory samples. First, we predicted these trajectories using a traditional LSTM model. Among such a large number of samples, only 159 trajectory samples showed significant deviation of prediction error from the real trajectory, which were mainly distributed in highly dynamic and complex traffic scenarios, such as sharp turns, sudden acceleration, and emergency braking. The results show that the LSTM model has good modeling ability in dealing with such time-series data as vehicle trajectories, and can cover most of the regular traffic behaviors. However, the traditional LSTM still shows a certain lack of robustness in the face of sudden and drastic nonlinear changes in the trajectory, especially the lag in the response to the recent state changes, which leads to a significant increase in the prediction bias.

Encouragingly, the EMA-LSTM model proposed in this paper successfully predicts 58 trajectories out of these 159 difficult samples in which LSTM performs poorly, i.e., its prediction error is significantly lower than that of the LSTM model in these scenarios. As shown in Fig. 3, we selected the representative cases of the above difficult samples and predicted them using the LSTM and

Table 1: RMSE per Time step for Different Models

Model	Timestep 1 (m)	Timestep 2 (m)	Timestep 3 (m)	Timestep 4 (m)	Timestep 5 (m)
CS-LSTM	0.1027	0.2019	0.3153	0.4378	0.5668
SA-LSTM	0.1023	0.2027	0.3162	0.4359	0.5647
Vanilla LSTM	0.1025	0.1543	0.3294	0.4853	0.6313
EMA-LSTM (Ours)	**0.1002**	**0.1998**	**0.3124**	**0.4327**	**0.5591**

EMA-LSTM models respectively. The blue solid line in the figure indicates the real trajectory, the green solid line indicates the prediction result of the EMA-LSTM model, and the orange solid line indicates the prediction result of the LSTM model.

It can be observed that at the turning point of the trajectory or the position where the motion trend changes significantly, the predicted trajectory of LSTM shows a large deviation, or even completely inconsistent with the real trajectory trend. On the other hand, the EMA-LSTM model shows higher tracking ability, and its predicted path is closer to the real trajectory, which can accurately reflect the spatial dynamic changes of the vehicle. This is because EMA-LSTM introduces an exponential weighting mechanism when modeling historical trajectories, which makes the model pay more attention to the recent motion state, and effectively alleviates the problem of LSTM's rapid information decay when dealing with drastic changes.

In summary, the EMA-LSTM model not only outperforms the traditional LSTM in terms of overall prediction accuracy, but also shows stronger robustness and dynamic adaptability in complex trajectory scenarios, which further verifies the effectiveness and practical value of introducing the EMA attention mechanism.

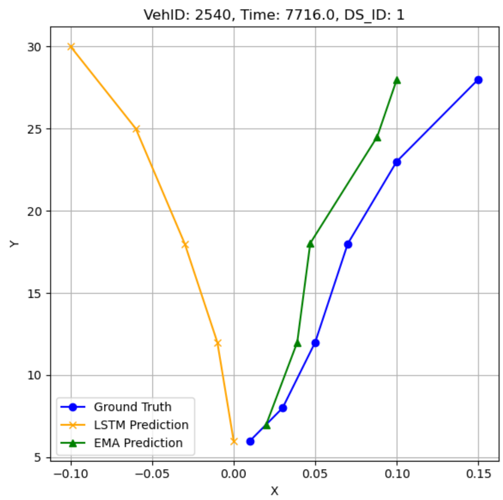

Fig. 3: Comparison of LSTM and EMA-LSTM Predictions in the Same Scenario

4.4. Comparison of Results

To confirm the EMA-Attention LSTM model's (subsequently termed EMA-LSTM) efficacy in trajectory forecasting, this study conducts comparative experiments with the traditional LSTM model, the vehicle trajectory prediction model that combines convolutional operations with LSTM networks (hereinafter referred to as CS-LSTM) [15], and the trajectory prediction model that combines spatial and attention mechanisms with LSTM networks (hereinafter referred to as SA-LSTM), were compared and experimented. The findings are depicted in Table 1, showcasing the RMSE metrics for various models over a 5 time step interval.

The analysis shows that the RMSE of EMA-LSTM outperforms the other comparison models at all time steps, demonstrating excellent robustness and high-precision prediction ability. Especially in the 3rd to 5th time steps, the error reduction is more significant, which further verifies the significant advantage of EMA-LSTM in the long-term prediction task.The RMSEs of CS-LSTM and SA-LSTM are close to each other in the first two time steps, but the errors gradually increase with the increase of time steps. This indicates that they are not capable of capturing long-term dependencies and are difficult to accurately model dynamic changes in complex traffic scenarios. The RMSE of the traditional LSTM model is considerably higher compared to the other models, especially at the farther time steps (e.g., the RMSE of the 5th time step reaches 0.6313 m), which verifies the necessity of introducing social interaction and attention mechanisms in the trajectory prediction task.

The significant advantages of EMA-LSTM over other models are reflected in the following aspects:

- The EMA module enhances model robustness and trajectory sensitivity by emphasizing critical time steps and spatial data while reducing noise.
- EMA-LSTM integrates the dynamic features of target vehicles and the social interactions of nearby vehicles, enhancing trajectory modeling in complex traffic situations and achieving superior performance in long-term predictions.
- The refined attention mechanism helps the model accurately detect vehicle interactions in complex traffic, significantly improving prediction accuracy.

Furthermore, to evaluate the computational efficiency of EMA-LSTM, we measured both training and inference time under the same hardware conditions (NVIDIA RTX 4070 Ti). The total training time for EMA-LSTM was approximately 2 hours and 10 minutes for 10 epochs, comparable to SA-LSTM (2 hours and 13 minutes). Additionally, the average inference time per sample was 0.0001 seconds, which remains within an acceptable range for real-time deployment. These results indicate that despite incorporating additional EMA and attention mechanisms, the proposed model maintains competitive computational efficiency.

5. CONCLUSION

The proposed EMA-Attention LSTM model enhances trajectory prediction robustness by integrating LSTM's long- and short-term dependency modeling with EMA's dynamic weighting, emphasizing key time steps and mitigating noise. The EMA mechanism employs a linear computational complexity, improving interpretability and reducing overhead. Experimental results demonstrate superior accuracy, efficiency, and robustness compared to traditional methods, highlighting its potential for complex traffic scenarios.

However, the model is currently limited to highway environments and relies solely on time series data. Future research could expand to diverse traffic scenarios and incorporate multimodal data (e.g., visual, radar) to enhance applicability and generalization.

In summary, the model enhances feature extraction and computational efficiency while improving real-time predictions and decision-making clarity. It simplifies computation without sacrificing prediction reliability, excelling in smart transportation and autonomous driving. By refining trajectory predictions, it reduces driving unpredictability, streamlines traffic patterns, and improves urban road management. Ultimately, it serves as a transformative framework for advancing intelligent transportation systems.Additionally, future work will explore representative scenarios where conventional models fail but EMA-LSTM provides accurate predictions, to better highlight its practical advantages.

REFERENCES

[1] F. Leon and M. Gavrilescu, "A Review of Tracking and Trajectory Prediction Methods for Autonomous Driving", *Mathematics*, Vol. 9, No. 6, p. 660, 2021.

[2] V. Bharilya and N. Kumar, "Machine Learning for Autonomous Vehicle's Trajectory Prediction: A Comprehensive Survey, Challenges, and Future Research Directions", *Vehicular Communications*, p. 100733, 2024.

[3] A. Quintanar, D. Fernández-Llorca, I. Parra, R. Izquierdo, and M. A. Sotelo, "Predicting Vehicles Trajectories in Urban Scenarios With Transformer Networks and Augmented Information", *Proc. IEEE Intelligent Vehicles Symposium (IV)*, pp. 1051–1056, 2021.

[4] T. A. Ranney, "Models of Driving Behavior: A Review of Their Evolution", *Accident Analysis & Prevention*, Vol. 26, No. 6, pp. 733–750, 1994.

[5] D. Chowdhury, L. Santen, and A. Schadschneider, "Statistical Physics of Vehicular Traffic and Some Related Systems", *Physics Reports*, Vol. 329, No. 4–6, pp. 199–329, 2000.

[6] G. Xie, H. Gao, L. Qian, B. Huang, K. Li, and J. Wang, "Vehicle Trajectory Prediction by Integrating Physics-And Maneuver-Based Approaches Using Interactive Multiple Models", *IEEE Transactions on Industrial Electronics*, Vol. 65, No. 7, pp. 5999–6008, 2017.

[7] C. G. Prevost, A. Desbiens, and E. Gagnon, "Extended Kalman Filter for State Estimation and Trajectory Prediction of a Moving Object Detected by an Unmanned Aerial Vehicle", *Proc. American Control Conference*, pp. 1805–1810, 2007.

[8] B. Kim, C. M. Kang, J. Kim, S. H. Lee, C. C. Chung, and J. W. Choi, "Probabilistic Vehicle Trajectory Prediction Over Occupancy Grid Map via Recurrent Neural Network", *Proc. IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pp. 399–404, 2017.

[9] F. Altché and A. de La Fortelle, "An LSTM Network for Highway Trajectory Prediction", *Proc. IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pp. 353–359, 2017.

[10] G. Xie, A. Shangguan, R. Fei, W. Ji, W. Ma, and X. Hei, "Motion Trajectory Prediction Based on a CNN-LSTM Sequential Model", *Science China Information Sciences*, Vol. 63, pp. 1–21, 2020.

[11] L. Lin, W. Li, H. Bi, and L. Qin, "Vehicle Trajectory Prediction using LSTMs with Spatial–Temporal Attention Mechanisms", *IEEE Intelligent Transportation Systems Magazine*, Vol. 14, No. 2, pp. 197–208, 2021.

[12] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Attention Based Vehicle Trajectory Prediction", *IEEE Transactions on Intelligent Vehicles*, Vol. 6, No. 1, pp. 175–185, 2020.

[13] S. Qiao, F. Gao, J. Wu, and R. Zhao, "An Enhanced Vehicle Trajectory Prediction Model Leveraging LSTM and Social-Attention Mechanisms", *IEEE Access*, 2023.

[14] U.S. Department of Transportation Federal Highway Administration, Next Generation Simulation (NGSIM) Program US-101 Videos, 2016. http://doi.org/10.21949/1504477

[15] N. Deo and M. M. Trivedi, "Convolutional Social Pooling for Vehicle Trajectory Prediction", *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1468–1476, 2018.