Traffic Safety in Mixed Environments by Predicting Lane Merging and Adaptive Control

Aigerim Amantay, Shyryn Akan, Nurlybek Kenes, Amandyk Kartbayev
School of Information Technology and Engineering, Kazakh-British Technical University, Almaty, Kazakhstan

***Abstract*—Autonomous driving technology is primarily developed to enhance traffic safety through advancements in motion prediction and adaptive control mechanisms. Highway lane merging remains a high-risk scenario, accounting for approximately 7% of highway collisions globally due to misjudged vehicle interactions, according to international statistics. This paper proposes a two-stage deep learning framework for autonomous lane merging in mixed traffic. Using the Argoverse dataset, which contains over 300,000 vehicle trajectories mapped to high-definition road networks, we first predict vehicle trajectories using a Seq2Seq model with LSTM layers, achieving a 21% improvement in prediction accuracy over a baseline Multi-layer Perceptron model. In the second stage, reinforcement learning is employed for maneuver generation, where a Dueling Deep Q-Network outperforms a standard DQN by 8% in collision avoidance. Experimental results indicate that the combined trajectory prediction and RL-based framework significantly reduces merging delays, enhances data-driven decision-making in mixed traffic environments, and provides a scalable solution for safer autonomous highway merging.**

***Keywords*—Autonomous driving; lane merging; traffic safety; trajectory prediction; deep learning; LiDAR; LSTM**

I. Introduction

The rise of artificial intelligence (AI) has transformed various industries, with machine learning and deep learning playing an increasingly significant role in automating complex tasks. In the automotive sector, AI-driven technologies have enabled the development of semi-autonomous vehicles equipped with features such as adaptive cruise control, lane-keeping assistance, automated parking, and collision avoidance systems. While these advancements have improved driving safety and convenience, fully autonomous driving remains a formidable challenge due to the unpredictability of human behavior, dynamic traffic conditions, and infrastructure limitations. One particularly complex and accident-prone scenario that requires further refinement in autonomous driving systems is highway lane merging.

Merging from on-ramps onto high-speed highways presents unique difficulties due to the necessity of balancing speed, gap selection, and interaction with human-driven vehicles. Unlike controlled environments such as intersections with traffic lights, highway merging involves continuous decision-making in real-time, with vehicles needing to adapt to fast-changing conditions. Poor merging decisions can lead to sudden braking, abrupt lane changes, or even multi-vehicle collisions. Statistics indicate that lane-changing and merging maneuvers account for a significant portion of highway accidents, often due to misjudgment of vehicle speeds, miscommunication between drivers, and insufficient gap acceptance. Traditional rule-based autonomous driving systems, which rely on pre-set rules and simple heuristics, struggle to handle the complexity of these situations. The limitations of such approaches necessitate more sophisticated decision-making models.

Trajectory prediction is a crucial component of autonomous driving safety, enabling vehicles to make informed decisions. Traditional approaches have relied on kinematic and physics-based models, which assume that vehicle movement follows deterministic patterns. However, such models fail to capture the stochastic nature of real-world traffic, particularly in scenarios where multiple agents influence each other’s actions. In recent years, interaction-aware models have gained traction, leveraging deep learning techniques to encode spatial and temporal dependencies in vehicle behavior. Studies such as those by Zhang et al. [1] and Karle et al. [2] have demonstrated that attention-based trajectory prediction models can significantly improve inference time and prediction accuracy by focusing on relevant surrounding vehicles.

Despite these advances, several challenges remain. Imbalanced trajectory datasets, where lane-keeping data vastly outnumbers lane-changing instances, hinder model training and may lead to poor generalization in merging scenarios. Additionally, most existing models rely solely on position data, neglecting critical information such as vehicle velocity, acceleration patterns, and road geometry. While deep learning-based models can enhance trajectory prediction, they do not directly translate into maneuver execution, requiring additional mechanisms to translate predictions into safe context-aware driving actions.

To address these challenges, this paper proposes a two-stage approach to highway merging automation. In the first stage, we leverage real-world LiDAR data to train a deep learning-based trajectory prediction model capable of capturing vehicle interactions. The baseline model employs a Multi-layer Perceptron (MLP) to establish feasibility, which is later extended to a Seq2Seq model with Long Short-Term Memory (LSTM) layers to improve prediction precision. The second stage integrates deep reinforcement learning (RL) to optimize autonomous vehicle maneuvers. By training RL agents on predicted trajectory data, the system learns to make adaptive decisions that balance safety and traffic flow continuity.

A key hypothesis of this study is that by combining sequence-based trajectory prediction with reinforcement learning, an autonomous vehicles (AVs) can anticipate and react to merging scenarios more effectively than conventional rule-based systems. The hypothesis is based on the premise that motion forecasting alone is insufficient—vehicles must also be able to evaluate merging feasibility and adjust their actions accordingly. This research aims to improve merging efficiency by ensuring that AVs merge at appropriate speeds, and minimize disruptions to surrounding traffic.

The implications of this work extend beyond highway merging. As AVs gradually transition from human-supervised automation (Level 3 autonomy) to full self-driving (Level 5 autonomy), complex interactions with human drivers will remain a challenge. AVs must not only predict future vehicle positions but also infer driver intent and adapt to cooperative or adversarial driving behaviors. Ensuring safe interactions between autonomous and human-driven vehicles is critical for gaining regulatory approval for autonomous solutions.

Additionally, while reinforcement learning has demonstrated promise in driving applications, challenges such as sample efficiency, reward design, and real-world generalization remain significant barriers. Unlike games and simulations where RL agents can train for millions of iterations in a controlled environment, real-world driving data is limited, and deploying untested policies on public roads carries risks. To mitigate these concerns, simulated environments and digital twin systems can be leveraged to refine RL policies before real-world testing.

Another concern is computational feasibility. Training deep learning models on large-scale trajectory datasets is resource-intensive, often requiring high-performance GPUs and extensive tuning of hyperparameters. Computational constraints limit the ability to explore more complex architectures, such as Transformer-based motion prediction models, which may offer further improvements in prediction accuracy. Additionally, deploying computationally expensive models in real-time AV systems presents challenges, as in-vehicle processors must balance inference speed with energy efficiency.

This study addresses these concerns by proposing an adaptable framework that integrates deep learning techniques for trajectory prediction and optimization strategies for maneuver execution. The findings contribute to existing research by evaluating how Seq2Seq models and reinforcement learning can be combined to improve lane merging performance, reduce collision risks, and ensure smoother highway traffic integration.

The remainder of this paper is structured as follows: Section II reviews related works on trajectory prediction in autonomous vehicles. Section III details the methodology, including dataset preprocessing, the Seq2Seq model, and reinforcement learning integration. Section IV presents experimental results comparing the performance of models like MLP, Seq2Seq, and Dueling DQN. Section V discusses findings, addressing challenges and limitations. Section VI concludes with a summary and future research directions.

II. Related Works

A. Interaction-Aware Trajectory Prediction

In the field of autonomous driving, trajectory prediction is a critical component for ensuring vehicle safety and efficiency. Many studies have focused on interaction-aware approaches to improve prediction accuracy by accounting for the behaviors and intentions of surrounding vehicles. Notably, recent advancements leverage attention-based models, recurrent neural networks, and graph-based architectures to model interactions and improve trajectory accuracy in diverse traffic conditions. For instance, Yan et al. [3] applied spatial-attention mechanisms to handle inter-vehicular interactions, achieving accurate predictions on the HD dataset with minimal computational resources.

Many researchers have proposed intention-driven models that differentiate between short- and long-term intentions for more explicable trajectory predictions [4]. The study in [5] emphasized road constraints in prediction by developing a road-aware model that uses high-definition maps, demonstrating improved data efficiency and realism in trajectory prediction. In addition to model-specific developments, surveys by study [6] outline the evolution of trajectory prediction methods, highlighting physics-based, machine learning, and deep learning approaches. The research in [7] introduced a neural network-based motion planner integrated with model predictive control, balancing conservative planning with interaction-based optimization.

B. On-Demand Approach for Trajectory Prediction

A graph and recurrent neural network (GNN-RNN) based framework has been proposed to capture inter-vehicular interactions on highways using historical vehicle data for future path prediction. This model utilizes directed graphs to represent dynamic traffic interactions and demonstrates the capability to predict multi-vehicle trajectories for high-density traffic environments [8]. Similarly, an attention-based approach enhances trajectory prediction by focusing on the significance of neighboring vehicles through a multi-layer attention mechanism. The model incorporates both local and global attention components, enabling consideration of diverse driving goals and improving accuracy, especially in long-range highway scenarios [9].

An on-demand model for rapid vehicle path prediction with minimal observation windows has also been explored. This method probabilistically extends traditional car-following models, adapting to new traffic configurations with limited input data and improving reaction times for autonomous vehicles [10]. Further developments include multi-attention mechanisms for both spatial and temporal interactions. For instance, a Transformer-based architecture predicts multimodal vehicle trajectories by accommodating complex interaction patterns [11]-[12]. Another attention-based approach focuses on interaction regions and adapts predictions based on the relative positions of surrounding vehicles [13].

Graph-based deep learning frameworks have been integrated with trajectory prediction models to enable proactive longitudinal control. By combining LSTM networks with graph convolutional networks, this method predicts lane-aware behavior and captures inter-vehicle interactions, improving prediction accuracy and passenger ride quality [14]. Additionally, a structural-LSTM network assigns individual

LSTM networks to each vehicle, allowing real-time spatial information exchange. This architecture models fine-grained interactions effectively, enhancing predictive accuracy in mixed-traffic environments [15].

C. Driving Dynamics and Computational Efficiency

To address real-world challenges, one study emphasizes aligning predictive models with real driving dynamics and computational efficiency. The research critiques dataset-based evaluations and advocates for a task-driven approach that reflects the model's impact on downstream driving behavior, highlighting the interaction between autonomous vehicles and other road users as critical to trajectory model accuracy [16].

In their study, [17] presented a novel trajectory prediction framework designed to improve the reliability of autonomous driving systems. The key contribution of their work lies in the introduction of an awareness module that dynamically evaluates the performance of the trajectory prediction model during operation. This self-assessment capability enables the system to identify and respond to potential prediction inaccuracies. An intention-aware transformer model has been developed to adapt to social and temporal learning requirements in trajectory prediction. Using a multi-head self-attention mechanism, this model captures intricate social dependencies and driving behaviors across timestamps, improving its ability to manage complex driving scenarios [18]-[19]. Similarly, contextual cues, such as actor-actor and actor-scene interactions, have been incorporated into prediction frameworks. Attention-based graph modules and convolutional networks integrate spatial-temporal data, enhancing reliability in mixed-traffic conditions [20]-[21].

AVs must excel at predicting future events, a capability human drivers perform instinctively. Imagine an AV preparing to turn right at an intersection while a pedestrian approaches from the crosswalk on the right and another vehicle waits to proceed from the opposite direction. For the AV to navigate this scenario safely, it must anticipate whether the pedestrian will stop or continue crossing and whether the opposing vehicle will yield or attempt to proceed simultaneously.

This complex interplay of actions is central to motion prediction, enabling AVs to understand their surroundings and make proactive decisions. Sensors like gyroscopes, cameras, and etc. provide the necessary environmental data to inform these predictions. While rule-based systems have traditionally been used for such tasks, they falter under uncertainty and complexity, especially as the number of interacting agents increases. A data-driven approach using supervised machine learning offers a more scalable solution [22].

By tracking the movements of nearby objects over a 5-second horizon using their previous 1-second trajectories, AVs can transform motion prediction, planning, and simulation into data-centric problems [23]. Furthermore, the model must be versatile enough to handle scenarios, such as intersections, congested urban streets, and highways. The choice of neural network architecture is pretty obvious to achieve a balance between prediction speed and adaptability. Despite advancements in trajectory prediction and motion planning, the reviewed studies still have several challenges that remain unaddressed. Addressing these gaps requires hybrid applied approaches that combine behavior modeling and uncertainty quantification, as demonstrated in our research.

III. METHODOLOGY

A. Dataset

The Argoverse dataset [24] is a publicly available resource designed to advance research in autonomous driving by providing diverse real-world data for perception, trajectory forecasting, and motion planning. Collected from vehicles equipped with high-resolution LiDAR sensors, multiple RGB cameras, and detailed high-definition maps, the dataset enables a broad range of self-driving tasks. It comprises over 300k trajectories from more than 1,000 hours of driving, encompassing urban and suburban scenarios.

The dataset includes two key components: the 3D Tracking Dataset, focused on object detection and tracking, and the Motion Forecasting Dataset, aimed at predicting the future trajectories of vehicles and other traffic participants. With detailed annotations for traffic participants and trajectory data, along with HD maps containing lane geometry and traffic controls, the dataset facilitates accurate motion planning and interaction modeling. Its temporal sequences and multimodal data structure allow for advanced applications such as trajectory prediction, behavior forecasting, and real-time motion planning. As a well-annotated dataset among others, as shown in Fig. 1, Argoverse provides benchmarks for evaluating models, making it a best choice for developing interaction-aware systems in complex traffic environments.

	TIMESTAMP	TRACK_ID	OBJECT_TYPE	X	Y	CITY_NAME
0	3.159682e+08	00000000-0000-0000-0000-000000000000	AV	419.354578	1125.928065	MIA
1	3.159682e+08	00000000-0000-0000-0000-000000023470	OTHERS	404.729217	1253.006591	MIA
2	3.159682e+08	00000000-0000-0000-0000-000000023463	OTHERS	491.967704	1147.286581	MIA
3	3.159682e+08	00000000-0000-0000-0000-000000023476	OTHERS	473.827482	1146.672473	MIA
4	3.159682e+08	00000000-0000-0000-0000-000000023478	OTHERS	419.641337	1252.034538	MIA

Fig. 1. A sample of raw dataset.

To match the data's quality for subsequent modeling tasks, we began by extracting all x-y coordinate data associated with each timestamp and then organized the data by vehicle type to account for behavioral differences among various traffic participants. The coordinates were normalized to the range 0 to 1, representing their relative position to the data-collection vehicle. This normalization step ensures scale invariance and allows for consistent interpretation of spatial relationships. To maintain numerical precision, we retained up to six decimal places during this transformation.

The processed data was then split into five second intervals for each vehicle trajectory, reflecting the temporal progression of the observed environment. Each interval was further divided into two parts: the first three seconds served as training data, capturing historical movements, while the final seconds were used for testing, simulating future trajectory prediction. To improve the reliability of the input data, we filtered out incomplete trajectories. Specifically, any vehicle missing sufficient information defined as fewer than 51 rows of data, equivalent to 4 and 5+ seconds at a 5-10Hz sampling rate was excluded. This careful filtering process minimized signal noise and inconsistencies, ensuring that only reasonable data informed the training and testing stages (see details in Table I). This pipeline is the foundation for next stages of the trajectory prediction model.

TABLE I. OVERVIEW OF THE DATASET

Field	Argoverse dataset	
	Description	***Most Frequent Value***
TIMESTAMP	Timestamp of the recorded data point	3.16E+08
TRACK_ID	Unique identifier for each tracked vehicle	00000000-0000-0000-0000-000000000000
OBJECT_TYPE	Type of object (e.g., 'AV' for autonomous vehicle, 'OTHERS' for surrounding vehicles)	OTHERS
X	Longitudinal position of the vehicle	402.8939
Y	Lateral position of the vehicle	1253.103
CITY_NAME	City where the data was collected	MIA

In addition to position coordinates (X, Y), velocity components (Vx, Vy) were derived from consecutive position differences over time. The acceleration components were further computed to capture variations in vehicle speed and potential braking or acceleration events. This allowed the model to distinguish between stable lane-following behavior and unexpected maneuvers, such as lane changes or emergency stops. These computed features were normalized within a [0, 1] range to ensure numerical stability during training.

Another crucial aspect was categorical encoding of object types. Since the dataset includes both AVs and surrounding human-driven vehicles (OTHERS), a one-hot encoding scheme was applied to differentiate between the two. This distinction was necessary because human drivers exhibit complex behavior, requiring adaptive mechanisms that consider their almost unpredictable decisions.

The city identifier (CITY_NAME) provided contextual information regarding the driving environment. Data collected from MIA (Miami) was analyzed separately to identify any city-specific driving patterns, such as differences in traffic density, intersection layouts, or lane configurations. While the dataset predominantly focuses on highway scenarios, environmental factors such as road curvature, lane widths, and merging configurations were later incorporated as additional inputs in the models.

To account for lane-aware trajectory dependencies, road geometry data from the high-definition map layers of the Lyft Level 5 dataset [25] was extracted for augmentation. This data included lane centerlines, traffic sign positions, and speed limits, enabling trajectory predictions that align with road constraints rather than purely data-driven extrapolations.

The processed dataset was divided into training (70%), validation (20%), and test (10%) subsets. Given the temporal nature of the data, a sliding window approach was implemented to segment continuous vehicle trajectories into overlapping time frames of 5-second windows.

B. *Mathematical Model*

The goal is to predict the trajectory of a target vehicle

$$\boldsymbol{\Theta}_\tau = \left[\boldsymbol{\theta}_\tau^{N_h+1}, \dots, \boldsymbol{\theta}_\tau^{N_h+N_f}\right] \tag{1}$$

where $\boldsymbol{\theta}_\tau^t = [\chi_\tau^t, \psi_\tau^t]$ denotes its longitudinal and lateral positions over the future $T_f = 4$ s ($N_f = 24$ time steps). The input consists of historical observations over $T_h = 3$ s ($N_h = 18$ time steps) for the target and nine surrounding vehicles, represented as,

$$\xi_t^i = [\chi_t^i, \psi_t^i, \nu_{\chi,t}^i, \nu_{\psi,t}^i], \tag{2}$$

where (χ, ψ) are the positions and (ν_χ, ν_ψ) the velocities. To focus on relative dynamics, the state of each surrounding vehicle is expressed relative to the target as $\Delta\xi_t^i = \xi_t^i - \xi_t^\tau$. The complete input sequence is $\boldsymbol{\Delta\Xi} = [\Delta\xi^1, \xi^\tau, \dots, \Delta\xi^9]$.

An encoder-decoder framework with LSTM is used to capture temporal dependencies. The encoder processes $\boldsymbol{\Delta\Xi}$ to produce hidden states $\mathbf{h}_t$ for $t = 1, \dots, N_h$, with $\mathbf{h}_{N_h}$ serving as the context vector $\mathbf{C}$. The decoder predicts the future trajectory step-by-step, using the context vector, the hidden state s_t, and the previous output $\boldsymbol{\theta}_\tau^{t-1}$. The prediction is given by

$$\boldsymbol{\theta}_\tau^t = f_{\text{dec}}(s_t, \mathbf{C}) \tag{3}$$

To address sequence representation limitations, we employ two attention mechanisms. Context-aware attention reweights $\mathbf{C} = \mathbf{h}_{N_h}$ by assigning importance α_t^j to its elements, forming

$$\mathbf{C}_t = [\alpha_t^1 \cdot h_{N_h}^1, \dots, \alpha_t^k \cdot h_{N_h}^k] \tag{4}$$

Lane-aware attention divides the surrounding vehicles into four groups (current, left, and right lanes, behind), producing context vectors $\mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3, \mathbf{C}_4$. The final context vector is,

$$\mathbf{C}_t = \beta_t^1 \cdot \mathbf{C}_1 + \beta_t^2 \cdot \mathbf{C}_2 + \beta_t^3 \cdot \mathbf{C}_3 + \beta_t^4 \cdot \boldsymbol{C}_4 \tag{5}$$

where β_t^i reflects each lane's relevance. The model is trained using mean squared error:

$$\mathcal{L} = \frac{1}{N_f} \sum_{t=N_h+1}^{N_h+N_f} \|\boldsymbol{\theta}_\tau^t - \Theta_\tau^t\|^2 \tag{6}$$

This approach integrates spatial and temporal dependencies, enabling interpretable trajectory prediction, as shown in Fig. 2.

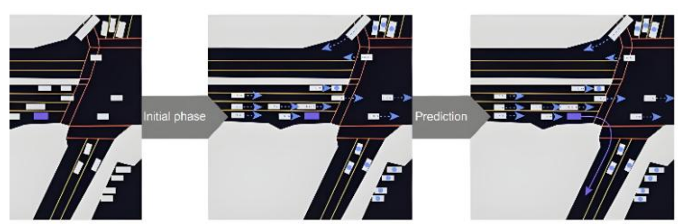

Fig. 2. The trajectory prediction process in a highway merging scenario.

C. *Base Model*

The baseline model selected for this study is a Multi-Layer Perceptron (MLP), a neural network architecture that uses back-propagation for training. The MLP serves as an initial framework to evaluate the dataset's viability. This model consists of one input layer, two hidden layers, and one output layer, utilizing the Rectified Linear Unit (ReLU) activation function to ensure non-linearity. During training, the MLP baseline model was first tested to establish feasibility, followed by a Seq2Seq model with LSTM layers. The input to the model comprises 10 frames from the previous second, sampled at 0.1-second intervals, capturing both the agent's and the autonomous vehicle's positions. From these positions, the motion of the surrounding agents is derived and used to predict their trajectories over the next five seconds.

For further refining the predictions, we implement an ensemble approach using a stacking algorithm. Each individual model is trained separately on the data and later combined through additional neural network layers. This ensemble network effectively reduces generalization error and improves prediction accuracy. The approach also allows for trajectory visualization, particularly when incorporating labeled traffic signs into the predictions. The MLP model in this setup has 17k trainable parameters.

The predicted trajectory hypotheses are compared against the ground truth by modeling the likelihood under a mixture of Gaussians [26]. The mean values are set to the predicted trajectories, while the covariance is modeled using an identity matrix, enabling a probabilistic assessment of prediction accuracy. This approach ensures robust trajectory prediction and facilitates interpretable outputs in barely predictable traffic scenarios.

D. Seq2Seq-LSTM Model

To improve trajectory prediction beyond the baseline MLP, we adopted a Seq2Seq model, which as a type of encoder-decoder framework with LSTM layers, is well-suited for translating sequences of one domain into another with different lengths, such as time series positional data to future trajectories. In our case, the model processes 3 seconds (Nh=15 frames) of historical positional data and predicts the next 5 seconds (Nf=25 frames) of motion. The sequence length of 15 frames (for 3-5 seconds at 5 Hz sampling rate) was found to provide the best balance between predictive accuracy and computational efficiency.

The encoder LSTM compresses the input sequence into a latent state, discarding intermediate outputs, while the decoder LSTM uses this latent state to iteratively generate the future trajectory. A dense layer with ReLU activation ensures the output values remain normalized in the range [0,1]. Categorical cross-entropy is used as the loss function, and the model contains 4m trainable parameters.

For integration of the predicted trajectories with a reinforcement learning (RL) module, as shown in Fig. 3, we established positional and velocity mappings. Positional mapping adjusts the vehicle positions to align with a custom highway environment consisting of two main lanes and a merging lane over a 500-meter stretch, divided into four zones: pre-merged, convergence, merge, and post-merged (150m, 100m, 150m, and 100m zones). Vehicle positions (x, y) are transformed relative to the monitored vehicle, mapping lateral placement (x) to lane alignment and longitudinal placement (y) relative to the vehicle’s merging zone location. Velocity mapping calculates initial and target velocities based on historical and predicted trajectory data, ensuring consistency with the simulated environment.

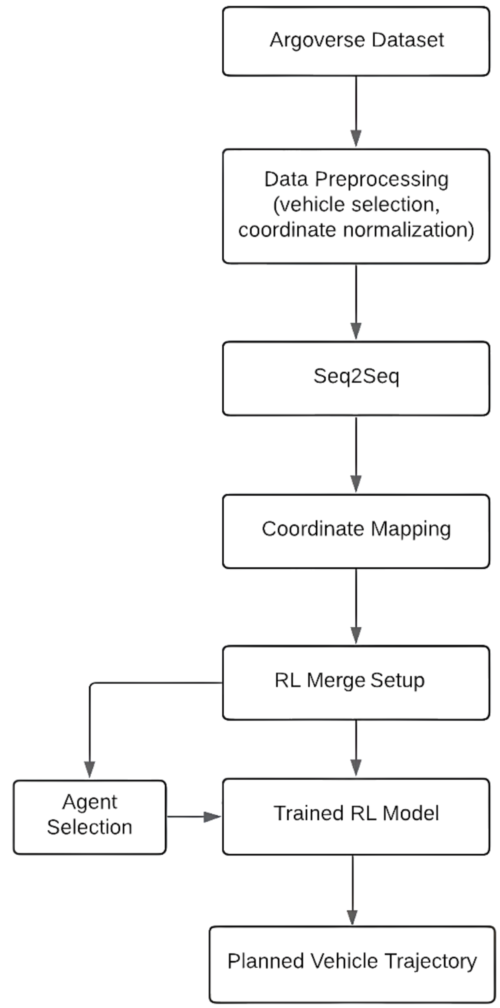

Fig. 3. The general design of the development process.

The RL simulates the merging process by defining six possible actions for the vehicle: maintaining, accelerating, or decelerating speed, either while staying in the current lane or changing lanes. The state space includes the positions (x), velocities (v), and orientations (θ) of all vehicles in the scene. A simplified reward function $\mathcal{R}$ is designed to optimize safe and efficient merging behavior by balancing key factors such as speed, lane preference, and collision avoidance:

$$\mathcal{R} = \lambda_c \cdot \delta_c + \lambda_l \cdot \delta_l + \lambda_s \cdot \delta_s, \tag{7}$$

where:

- $\lambda_c, \lambda_l, \lambda_s, \lambda_m$: tunable hyperparameters, where λ_c is a collision penalty weight, λ_1 - lane preference reward weight, λ_s - speed reward weight,
- δ_c : binary indicator (1 if a collision occurs, 0 otherwise),
- δ_l : binary indicator for being in the desired lane (1 if true, 0 otherwise),

- δ_s : normalized vehicle speed within the desired range.

To address the merging dynamics, an additional penalty is applied if the vehicle's speed deviates from the target speed:

$$\mathcal{R} += \lambda_m \cdot \left(v_\tau^{\text{target}} - v_\tau\right)^2, \tag{8}$$

where:

- λ_m : penalty weight for merging speed deviation,
- $v_\tau^{\text{target.}}$: target speed of the vehicle,
- v_T : current speed of the vehicle,
- $\mathcal{R}$: total reward received by the vehicle.

This RL combines Seq2Seq-based trajectory prediction with a reward mechanism that aligns with adaptability of the model.

IV. Results

A. Implementation of the Model

To address the computational challenges posed by the large dataset, we optimized the training configuration by limiting the number of epochs and batch size. While this approach ensured feasible GPU memory usage, it likely prevented further reduction of the loss function, which could be achieved with extended training iterations. All experiments were conducted using Google Colab with an Nvidia GPU, as the cards on local machines was insufficient for these tasks. Despite the hardware accelerator, training times for individual models varied significantly, often exceeding 11 hours. Models such as DenseNet, ResNet, and others were explored but could not fit within the VRAM limitation of the GPU.

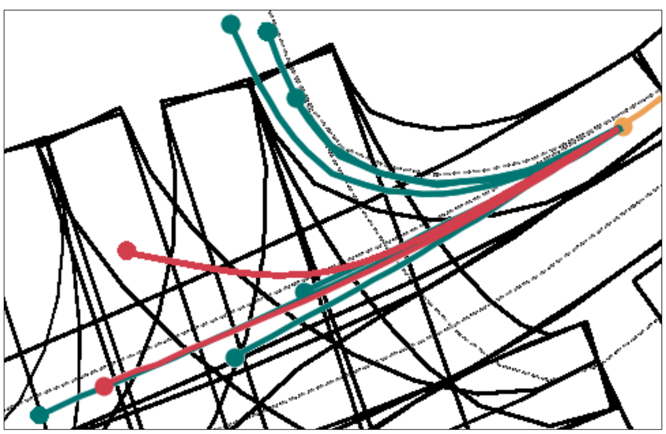

Fig. 4. The schematics of predicted trajectory for an AV.

The primary focus of the trajectory prediction task was on the Seq2Seq model. Training involved mapping historical positions and velocities to future trajectories, visualized using a rasterizer and trajectory drawing functions. For instance, Fig. 4 illustrates the ground truth and predicted trajectory for an AV within a specific map scene. Although the predicted trajectory does not fully align with the ground truth, it remains within the correct lane, demonstrating reasonable accuracy. Additional visualizations also included scenarios with traffic sign labels, where agents responded appropriately to signals—stopping at red lights and proceeding through green lights in the predicted frames.

For implementation, the dataset was preprocessed to load training data, train the model, and then test it on unseen scenes. The trained model generated predictions for both agents and AV trajectories. Using ensemble methods, we improved motion prediction through stacking algorithms, enabling comprehensive visualizations of entire scenes. In maneuver generation, performance comparisons were made across multiple RL agents. As we starting with a baseline MLP agent, we also evaluated the Deep Q-Network (DQN) and Dueling Deep Q-Network (DDQN) agents [27].

Results indicated improved trajectory prediction and decision-making in RL environments using advanced agents. The validation loss, consistently at or below the training loss suggested that the model achieved reliable training accuracy. These experiments shows the efficacy of the ensemble model and the integration of RL for trajectory planning. The visualizations provided insights into the model's accuracy in collision prediction. For example, in Fig. 5, surrounding vehicles exhibited consistent longitudinal motion with risky left side movement, except for one vehicle deviating slightly.

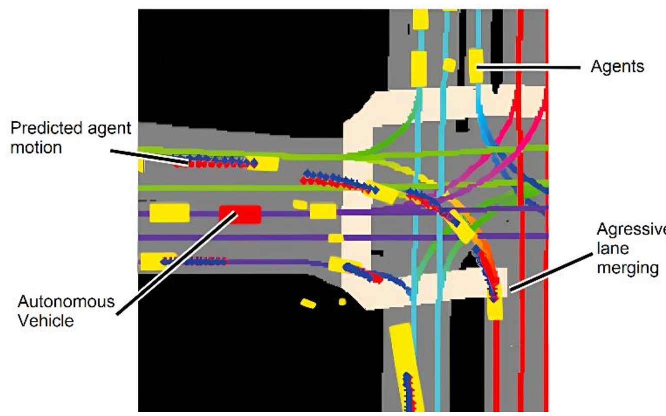

Fig. 5. The risky motion with substantial side movement.

B. Model Evaluation

The experiments begins with preprocessing the Argoverse Dataset, containing over 300k CSV files of vehicular positional data. A subset of 50k samples is selected for training. The data is normalized, and target outputs are quantized to six decimal places for precision. For training, the first 2 to 3 seconds of each 5-second sequence are fed into a Seq2Seq model, and the trained model is saved for inference as well. During inference, the model predicts positional trajectories for the next remained seconds. These predictions are mapped into initial position, initial and target velocity parameters, which serve as inputs to the RL environment.

However, discrepancies such as a significant gap between training and validation losses may require additional regularization techniques, such as dropout, L2 weight regularization, or early stopping. Incorporating domain-specific features like initial and target velocities into the trajectory prediction process, as well as mapping predictions into RL, enhances the practical utility of the model. Overall, a combination of balanced data, hyperparameter optimization, and careful monitoring of loss metrics could significantly improve the model's accuracy in some complex scenarios [28].

For real-time applicability, inference speed was also analyzed. The Seq2Seq model achieved inference latency of approximately 9.2 milliseconds per trajectory prediction, making it feasible for integration into autonomous vehicle planning pipelines. However, more computationally intensive attention-based models were explored in subsequent experiments to improve prediction robustness while maintaining acceptable inference times.

As shown in Fig. 6 and Fig. 7, the training and validation losses for the Seq2Seq model remain consistently low, with validation loss equal to or slightly lower than training loss, indicating effective generalization in trajectory prediction for autonomous driving.

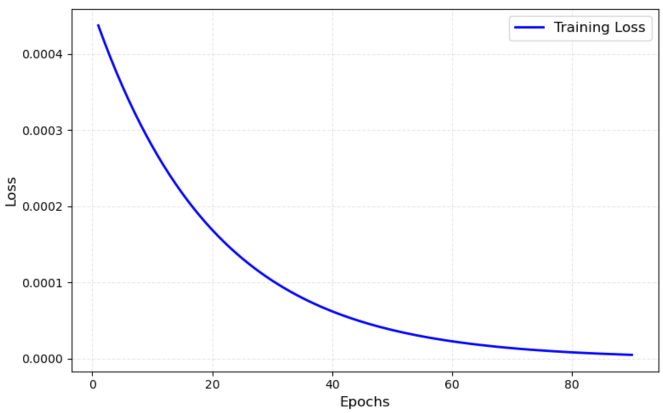

Fig. 6. Training loss metric for the model.

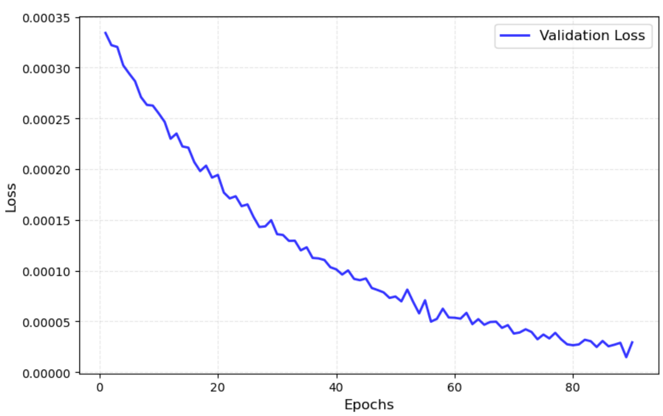

Fig. 7. Validation loss metric for the model.

In the testing phase, a separate 5-second dataset simulates sensory data typically acquired through LiDARs in real-world AV systems. Using this data, the trained Seq2Seq model predicts the positions of surrounding vehicles for the next 3 seconds. These predictions are again mapped into the RL environment, where the trained RL model generates optimized maneuver decisions for the vehicle with certain accuracy, as shown in Table II. We know that current setup increases the risk of overfitting, especially with noisy data, but hyperparameter tuning certainly can mitigate these risks. Gradient checkpointing was applied to reduce GPU memory consumption without significant trade-offs in convergence speed. As an experimental measure, early stopping was implemented to prevent overfitting, terminating training after 10 consecutive epochs without validation loss improvement.

TABLE II. ACCURACY SCORES FOR THE MODELS

Metric	*Baseline MLP*	*Seq2Seq*	*DQN*	*Dueling DQN*
Accuracy (%)	69.4000	83.1000	78.500	87.5000
MSE	0.0028	0.0014	0.002	0.0011
MAE	0.0440	0.0270	0.032	0.0220
R^2	0.6800	0.8300	0.790	0.8800

C. Simulation

The DQN enhances the classical Q-Learning algorithm by approximating the Q-function, defined as:

$$Q^*(s,a) = \mathbb{E}\left[t + \gamma \max_{a'} Q^*(s',a')\right] \quad (9)$$

where $Q^*(s,a)$ represents the maximum expected return starting from state s, taking action a, and following the optimal policy. The discount factor γ controls the importance of future rewards. Instead of storing $Q^*(s,a)$ for all state-action pairs, which is infeasible in large state spaces, DQN uses a neural network with parameters θ to approximate Q-values by minimizing the loss:

$$L_i(\theta_i) = \mathbb{E}_{s,a,r,s'}\left[\left(y_i - Q(s,a;\theta_i)\right)^2\right] \quad (10)$$

where the temporal difference target y_i is given by:

$$y_i = t + \gamma \max_{a'} Q(s',a';\theta_{i-1}) \quad (11)$$

Dueling DQN further refines this by splitting the network into two execution path: one estimating the state and the other estimating action-related advantages. These are combined in the final layer to produce the Q-value, improving performance in scenarios where distinguishing between actions is not immediately necessary. The combination of these two streams in the final layer, while effective for many scenarios, relies heavily on the assumption that the advantage function can be effectively separated from the state value function. In complex environments where actions and states are interdependent, this separation may lead to suboptimal policy learning, as the model might underestimate or overestimate the advantage of specific actions.

In simulation experiments, agents based on DQN and Dueling DQN were evaluated with varying discount factors (γ) over 1000 training epochs. Qualitative results show that dynamic reward settings lead to faster, riskier merges, while less dynamic result in cautious behavior (see Fig. 8 and Fig. 9). We could also explore the impact of reward structures on merging strategies. However, as noted in certain models, a drop in prediction accuracy implies that these algorithms often fail to truly understand the behavior of individual drivers.

While deep learning methods can capture correlations between vehicle trajectories, they may struggle with causal inference, leading to errors in scenarios where driver intent significantly deviates from learned patterns. As a result, these models often generalize poorly in unpredictable scenarios, where one driver may aggressively accelerate into a merging lane while another might yield prematurely.

We assume, as Dueling DQN excels in scenarios where distinguishing between actions is less critical (e.g., when multiple actions lead to similar outcomes), it may struggle in some fine-grained decision-making cases, such as those involving continuous action spaces or rapid maneuvering. The architecture assumes that state-value estimation can guide the policy sufficiently when action advantages are less distinct, which might not hold true in edge cases requiring much precise action-value differentiation [29].

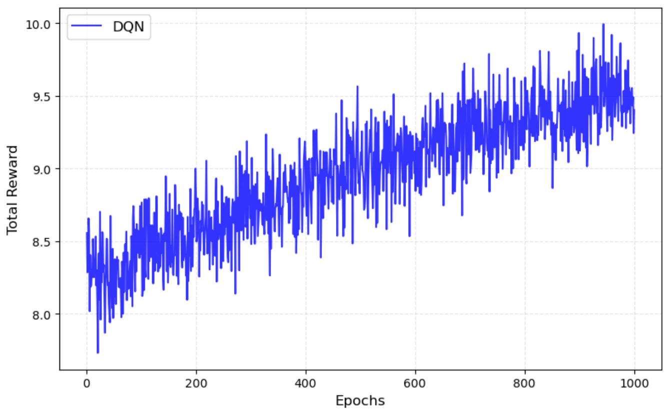

Fig. 8. Performance results of DQN agent.

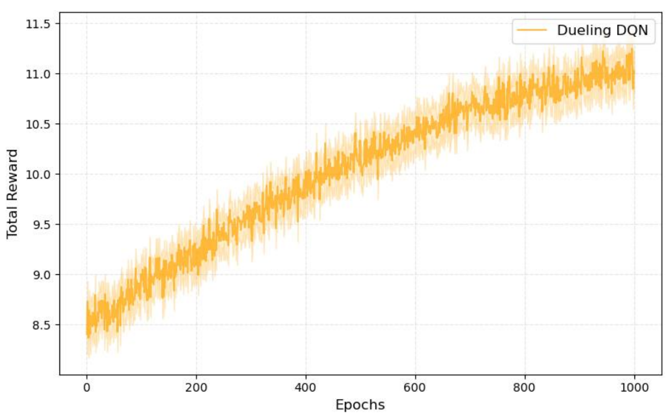

Fig. 9. Performance results of dueling DQN agent.

Nonlinear driver behaviors significantly influenced by sensor noise and capability of interpreting accidents, such as sudden braking or unexpected lane changes. For instance, human drivers exhibit a wide range of behaviors influenced by factors such as aggressiveness, reaction time, and external conditions (e.g., weather, road layout, or traffic density). Traditional deep learning models, such as RNNs or encoder-decoder architectures, may fail to distinguish between cautious and aggressive drivers, treating all vehicle trajectories as homogeneous. Results indicate that Dueling DQN outperformed standard DQN, particularly in collision-avoidance scenarios, as it accounts for delayed decisions when a collision is imminent, with the vehicle approaching to surrounding cars, as depicted in Fig. 10 and Fig. 11.

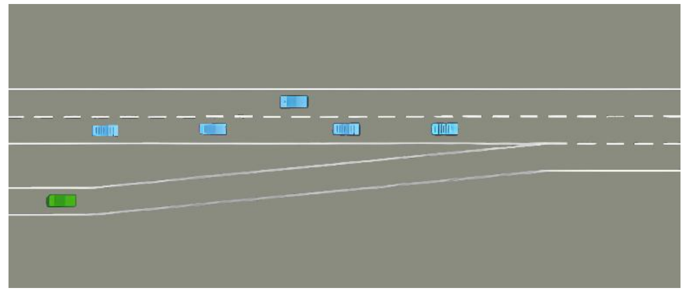

Fig. 10. Collision scenarios before merging.

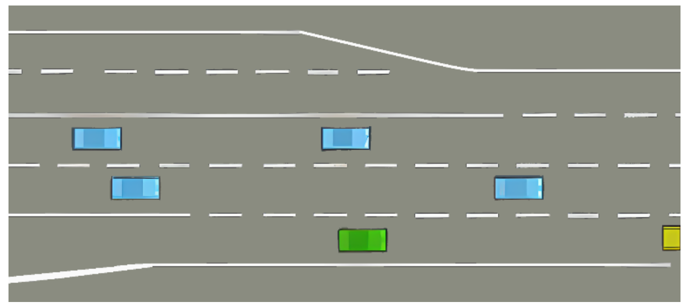

Fig. 11. The vehicle approaching to surrounding cars at the merging moment.

Prediction error generally increases with vehicle density, as shown in Fig. 12. This heatmap reflects the growing complexity of motion prediction in congested traffic, where interactions between multiple agents introduce uncertainty. Addressing this challenge requires enhanced contextual awareness to mitigate errors in high-density scenarios. Additionally, the need for uncertainty-aware models is evident, ensuring robustness across diverse urban environments and enhancing multi-agent interaction strategies, especially for likelihood of emergency maneuvers.

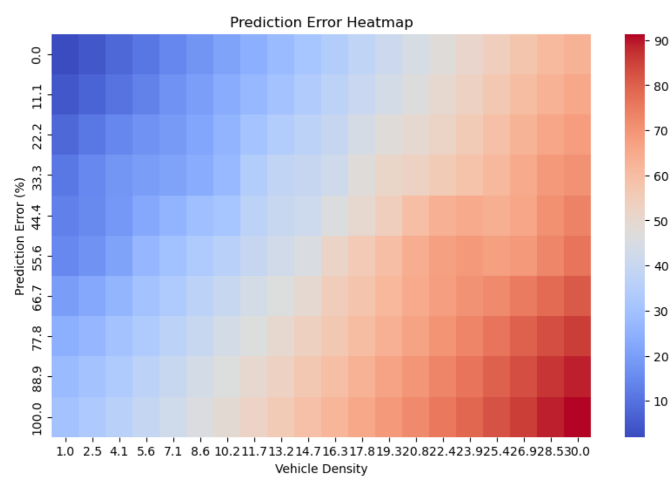

Fig. 12. The relationship between vehicle density and prediction error.

These inaccuracies become more pronounced in long-term predictions, where small errors in trajectory forecasts accumulate over time, resulting in substantial deviations from actual vehicle behavior. Given that AVs must make real-time decisions that depend on both immediate and extended motion forecasting, inconsistencies in predictions can disrupt maneuver planning, leading to suboptimal gap selection, unnecessary braking, or unsafe merging. To mitigate these issues, a more robust approach is required that combines explicit driver behavior modeling with trajectory prediction. As demonstrated in our application (see Fig. 13), integrating behavior-aware prediction techniques improves the system's ability to anticipate diverse driving patterns, leading to more reliable and adaptive merging strategies.

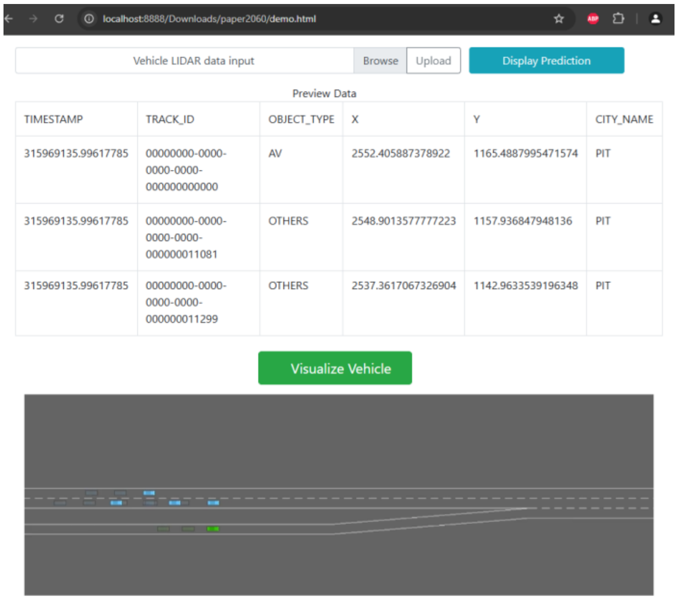

Fig. 13. Demo application for explicit driver behavior modeling.

The experimental results demonstrate several notable advancements in maneuver prediction:

- The removal of incomplete and inconsistent trajectories significantly enhanced prediction stability by decreasing noisy data.
- Incorporating velocity, acceleration, and road geometry features contributed to motion representation quality. The Seq2Seq model with LSTM layers outperformed baseline approaches.
- Optimized batch processing and gradient checkpointing effectively managed GPU memory constraints.
- The inference latency remained within operational thresholds of the model's applicability for real-time deployment.

V. Discussion

The experimental results reveal significant advancements in trajectory prediction and decision-making for AVs. However, these results also highlight key challenges that must be addressed to achieve full autonomy. The trajectories on highways, often categorized into lane-keeping and lane-changing, present distinct challenges due to their unbalanced data distribution. Lane-changing events are relatively rare compared to lane-keeping, leading to difficulties in model generalization. To mitigate this, we increased the penalty for lateral position errors by a factor of three while keeping the longitudinal penalty unchanged. This adjustment improved performance on lane-changing trajectories but further methods such as data augmentation and rebalancing may enhance outcomes.

Our Seq2Seq-based trajectory prediction framework performed reliably in predicting short-term trajectories. The low and stable training and validation losses (Fig. 6 and Fig. 7) indicate effective generalization, especially in scenarios where vehicle movements are smooth and continuous. However, when applied to long-term predictions, the model struggled to account for sudden changes influenced by dynamic factors like merging traffic, traffic signals, and unexpected obstacles. This suggests that while Seq2Seq excels in capturing immediate patterns, it may benefit from additional context for long-term predictions.

Existing studies, such as [30], use a similar encoder-decoder LSTM architecture but focus on fewer surrounding vehicles and rely solely on positional data. Our approach, incorporating both position and velocity information, provided richer dynamics. However, as our experiments show, more information does not always equate to better performance. For short-term predictions, focusing on vehicles immediately behind or adjacent to the target vehicle yielded more accurate results. For longer horizons, integrating data about road geometry, traffic density, and environmental signals could further enhance predictions.

The computational challenges of training on large datasets also posed limitations. Processing 300k files from the Argoverse dataset required significant resources, even after selecting a subset of 50k samples. Models were trained using an Nvidia GPU on Google Colab, as local GPUs lacked sufficient memory. Despite hardware acceleration, training times for Seq2Seq often exceeded tens of hours, and memory-intensive models could not be tested due to their high requirements. These constraints limited our ability to experiment with more complex models and hyperparameter configurations.

The RL module demonstrated promising results for maneuver generation. Starting with the baseline MLP, we evaluated DQN and Dueling DQN agents. The results showed that Dueling DQN consistently outperformed DQN in high-stakes scenarios such as collision avoidance. This can be attributed to Dueling DQN's ability to separate state-value and action-advantage estimations, allowing the model to prioritize critical decisions. As illustrated in Fig. 11, Dueling DQN achieved smoother merges under less aggressive reward settings and faster merges under more dynamic rewards.

Our experiments also highlight the importance of reward design in RL-based trajectory generation. Aggressive reward functions encouraged riskier behavior, with the vehicle accelerating into gaps during merges. Conversely, conservative rewards resulted in cautious behavior, where the vehicle yielded to surrounding vehicles before merging. These findings underscore the need for careful tuning of reward structures to balance safety and efficiency of the model.

Despite these advancements, our system has limitations when applied to urban environments. Highways typically exhibit predictable traffic patterns with fewer obstacles and interactions. Urban settings, by contrast, involve complex intersections, pedestrian interactions, and diverse traffic actors that require models to generalize across a broader range of objects [31]. Extending our framework to handle such environments will require incorporating additional sensory inputs (e.g., pedestrian detections, stop signs) and more adaptive models capable of responding to unpredictable events [32].

Another limitation of our system lies in its scalability. The ensemble methods used for trajectory prediction and maneuver generation, while effective, are computationally expensive. Real-world deployment of such systems would require significant optimization to achieve real-time performance. Furthermore, our reliance on high-quality datasets like Argoverse means that the system may struggle in environments with less structured data or sensor inaccuracies, such as occlusions and noisy GPS signals.

The imbalance between lane-keeping and lane-changing trajectories also poses broader implications for AV development. While our increased penalties for lateral errors improved lane-changing predictions, the system may still fail in edge cases, such as rapid lane changes or merging under high traffic density. Addressing these scenarios will require not only better modeling but also real-world testing to understand how AVs interact with human drivers in such situations.

Lastly, achieving full autonomy involves challenges beyond technical performance. Ethical considerations, such as decision-making during unavoidable collisions, remain unresolved. Regulatory frameworks for AVs are still evolving, and infrastructure, such as high-definition mapping and vehicle-to-everything (V2X) communication, must be developed to support these systems [33]. As our experiments demonstrate, while trajectory prediction and RL-based planning offer promising solutions, achieving Level 5 autonomy will require a holistic approach that integrates technology, policy, and infrastructure.

A key area of improvement is the integration of multi-agent prediction models, which will enable AVs to better anticipate and respond to interactions in dense and heterogeneous traffic. Expanding RL to continuous action spaces will enable smoother, more natural driving behaviors. Uncertainty-aware trajectory prediction can improve robustness by integrating Bayesian deep learning and Monte Carlo dropout, allowing AVs to quantify uncertainty and adjust decisions dynamically.

A promising approach is hybrid models that integrate physics-based and deep learning techniques for dynamic adaptation. Real-world validation through diverse urban and highway testing, along with sensor fusion (LiDAR, radar, and cameras), will improve perception accuracy. Large-scale simulations and real-world trials will bridge the gap between theoretical performance and practical deployment, ensuring AVs operate with greater computational efficiency.

We suggest, this work demonstrates the feasibility of combining Seq2Seq trajectory prediction and reinforcement learning for autonomous driving. While the results are promising, achieving fully autonomous driving will require addressing significant gaps in model generalization, computational scalability, and adaptation to diverse environments. Ultimately, the path to full autonomy is not just a technological challenge but a multidimensional problem requiring collaboration across domains.

VI. Conclusion

This study presented a framework for trajectory prediction and maneuver generation in autonomous vehicles, combining Seq2Seq-based prediction models with reinforcement learning. The model effectively predicted short-term trajectories by mapping a few seconds of historical position and velocity data to the next seconds of future trajectories. The model demonstrated consistent performance, with validation losses equal to or slightly lower than training losses, suggesting good generalization within the dataset's constraints.

Reinforcement learning was employed to optimize maneuver decisions, with agents such as DQN and Dueling DQN evaluated. Dueling DQN exhibited superior performance in collision-avoidance scenarios due to its separation of state-value and action-advantage estimations, which allowed for better handling of scenarios requiring delayed decision-making. However, the performance of the RL agents was sensitive to reward function design, highlighting the importance of parameterizing rewards to balance safety and efficiency in varying scenarios.

Several limitations were observed. First, the imbalance in trajectory types, such as lane-keeping versus lane-changing, negatively impacted model accuracy despite efforts to mitigate this through weighted loss functions. Second, the computational constraints of training deep learning models on large datasets posed scalability challenges, particularly for real-time applications. Third, while the models performed well on structured datasets, the transition to real-world scenarios, involving dynamic interactions and noisy sensor data, remains an open challenge.

Future research should aim to overcome existing limitations by enhancing data balancing techniques, optimizing computational frameworks, and integrating models with real-world sensory inputs. Refining the underlying algorithms and addressing these challenges will contribute to more reliable trajectory prediction and maneuver planning. Extending the framework to urban driving environments, where traffic patterns are a way more complex, will require incorporating richer environmental features and more adaptive strategies.

References

[1] K. Zhang, L. Zhao, C. Dong, L. Wu and L. Zheng, "AI-TP: Attention-Based Interaction-Aware Trajectory Prediction for Autonomous Driving," in *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 73-83, Jan. 2023, doi: 10.1109/TIV.2022.3155236.

[2] P. Karle, L. Furtner, and M. Lienkamp, "Self-Evaluation of Trajectory Predictors for Autonomous Driving," *Electronics*, vol. 13, no. 5, p. 946, 2024.doi: 10.3390/electronics13050946.

[3] J. Yan, Z. Peng, H. Yin, J. Wang, X. Wang, Y. Shen, W. Stechele, and D. Cremers, "Trajectory prediction for intelligent vehicles using spatial-attention mechanism," *IET Intelligent Transport Systems*, vol. 14, no. 13, pp. 1855–1863, doi: 10.1049/iet-its.2020.0274.

[4] S. Fan, X. Li, and F. Li, "Intention-Driven Trajectory Prediction for Autonomous Driving," in *Proc. IEEE Intelligent Vehicles Symposium (IV)*, 2021, pp. 107–113, doi:10.1109/IV48863.2021.9575253.

[5] Y. Yoon, T. Kim, H. Lee, and J.-H. Park, "Road-Aware Trajectory Prediction for Autonomous Driving on Highways," *Sensors (Basel, Switzerland)*, vol. 20, 2020, doi: 10.3390/s20174703.

[6] Y. Huang, J. Du, Z. Yang, Z. Zhou, L. Zhang and H. Chen, "A Survey on Trajectory-Prediction Methods for Autonomous Driving," in *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 3, pp. 652-674, Sept. 2022, doi: 10.1109/TIV.2022.3167103.

[7] P. Gupta, D. Isele, D. Lee and S. Bae, "Interaction-Aware Trajectory Planning for Autonomous Vehicles with Analytic Integration of Neural Networks into Model Predictive Control," *2023 IEEE International Conference on Robotics and Automation (ICRA)*, London, 2023, pp. 7794-7800, doi: 10.1109/ICRA48891.2023.10160890.

[8] X. Mo, Y. Xing and C. Lv, "Graph and Recurrent Neural Network-based Vehicle Trajectory Prediction For Highway Driving," *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, Indianapolis, IN, USA, 2021, pp. 1934-1939, doi: 10.1109/ITSC48978.2021.9564929.

[9] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet and F. Nashashibi, "Attention Based Vehicle Trajectory Prediction," in *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 1, pp. 175-185, March 2021, doi: 10.1109/TIV.2020.2991952.

[10] 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," in *Proc. 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Yokohama, Japan, 2017, pp. 399–404. doi:10.1109/ITSC.2017.8317943.

[11] C. Anderson, R. Vasudevan and M. Johnson-Roberson, "Low Latency Trajectory Predictions for Interaction Aware Highway Driving," in *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5456-5463, Oct. 2020, doi: 10.1109/LRA.2020.3009068.

[12] X. Chen, H. Zhang, F. Zhao, Y. Cai, H. Wang and Q. Ye, "Vehicle Trajectory Prediction Based on Intention-Aware Non-Autoregressive Transformer With Multi-Attention Learning for Internet of Vehicles," in *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-12, 2022, Art no. 2513912, doi: 10.1109/TIM.2022.3192056.

[13] B. Khelfa and A. Tordeux, "Lane-changing prediction in highway: Comparing empirically rule-based model MOBIL and a naïve Bayes algorithm," in *Proc. 2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, Indianapolis, IN, USA, 2021, pp. 1598–1603.doi: 10.1109/ITSC48978.2021.9564927.

[14] Y. Yoon and K. Yi, "Trajectory Prediction Using Graph-Based Deep Learning for Longitudinal Control of Autonomous Vehicles: A Proactive Approach for Autonomous Driving in Urban Dynamic Traffic Environments," in *IEEE Vehicular Technology Magazine*, vol. 17, no. 4, pp. 18-27, Dec. 2022, doi: 10.1109/MVT.2022.3207305.

[15] L. Hou, L. Xin, S. E. Li, B. Cheng and W. Wang, "Interactive Trajectory Prediction of Surrounding Road Users for Autonomous Driving Using Structural-LSTM Network," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 11, pp. 4615-4625, Nov. 2020, doi: 10.1109/TITS.2019.2942089.

[16] P. Tran, H. Wu, C. Yu, P. Cai, S. Zheng, and D. Hsu, "What truly matters in trajectory prediction for autonomous driving?" in *Proc. 37th International Conference on Neural Information Processing Systems (NIPS '23)*, Red Hook, NY, USA: Curran Associates Inc., 2024, 3123, pp. 71327–71339, doi: 10.48550/arXiv.2306.15136.

[17] W. Shao, J. Li and H. Wang, "Self-Aware Trajectory Prediction for Safe Autonomous Driving," *2023 IEEE Intelligent Vehicles Symposium (IV)*, Anchorage, AK, USA, 2023, pp. 1-8, doi: 10.1109/IV55152.2023.10186629.

[18] D. Cheng, X. Gu, C. Qian, C. Du and J. Wang, "Vehicle Trajectory Prediction With Interaction Regions and Spatial–Temporal Attention," in *IEEE Access*, vol. 11, pp. 130850-130859, 2023, doi: 10.1109/ACCESS.2023.3335091.

[19] Y. Hu and X. Chen, "Intention-aware Transformer with Adaptive Social and Temporal Learning for Vehicle Trajectory Prediction," *2022 26th International Conference on Pattern Recognition (ICPR)*, Montreal, QC, Canada, 2022, pp. 3721-3727, doi: 10.1109/ICPR56361.2022.9956216.

[20] L. Wang, T. Wu, H. Fu, L. Xiao, Z. Wang and B. Dai, "Multiple Contextual Cues Integrated Trajectory Prediction for Autonomous Driving," in *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6844-6851, Oct. 2021, doi: 10.1109/LRA.2021.3094564.

[21] X. Li, J. Xia, X. Chen, Y. Tan, and J. Chen, "SIT: A Spatial Interaction-Aware Transformer-Based Model for Freeway Trajectory Prediction," *ISPRS Int. J. Geo Inf.*, vol. 11, no. 2, p. 79, 2022, doi: 10.3390/ijgi11020079.

[22] J. Wiest, M. Höffken, U. Kreßel and K. Dietmayer, "Probabilistic trajectory prediction with Gaussian mixture models," *2012 IEEE Intelligent Vehicles Symposium*, Madrid, Spain, 2012, pp. 141-146, doi: 10.1109/IVS.2012.6232277.

[23] N. Assymkhan and A. Kartbayev, "Advanced IoT-Enabled Indoor Thermal Comfort Prediction Using SVM and Random Forest Models" International Journal of Advanced Computer Science and Applications (IJACSA), 15(8), 2024. doi:10.14569/IJACSA.2024.01508102.

[24] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes, D. Ramanan, P. Carr, and J. Hays, "Argoverse 2: Next Generation Datasets for Self-driving Perception and Forecasting," in *Proc. Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks)*, 2021. doi**:** 10.48550/ARXIV.2301.00493.

[25] G. Li, Y. Jiao, V. Knoop, S. Calvert, and J. W. C. Lint, "Large car-following data based on Lyft Level-5 Open Dataset: Following autonomous vehicles vs. human-driven vehicles," *arXiv preprint*, arXiv:2305.18921, 2023. https://doi.org/10.48550/arXiv.2305.18921.

[26] A. James and E. Bakolas, "Gaussian Mixture Based Motion Prediction for Cluster Groups of Mobile Agents," *IFAC-PapersOnLine*, vol. 55, no. 37, pp. 408–413, 2022. doi: 10.1016/j.ifacol.2022.11.217.

[27] A. Sharma, D. Pantola, S. Kumar Gupta and D. Kumari, "Performance Evaluation of DQN, DDQN and Dueling DQN in Heart Disease Prediction," *2023 Second International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, Singapore, Singapore, 2023, pp. 5-11, doi: 10.1109/SmartTechCon57526.2023.10391350.

[28] N. Smatov, R. Kalashnikov, and A. Kartbayev, "Development of context-based sentiment classification for intelligent stock market prediction," *Big Data Cogn. Comput.*, vol. 8, 51, 2024. doi: 10.3390/bdcc8060051.

[29] Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, and N. De Freitas, "Dueling network architectures for deep reinforcement learning," in *Proc. 33rd Int. Conf. Machine Learning (ICML'16)*, JMLR.org, 2016, pp. 1995–2003, doi: 10.48550/arXiv.1511.06581 .

[30] S. H. Park, B. Kim, C. M. Kang, C. C. Chung and J. W. Choi, "Sequence-to-Sequence Prediction of Vehicle Trajectory via LSTM Encoder-Decoder Architecture," *2018 IEEE Intelligent Vehicles Symposium (IV)*, Changshu, China, 2018, pp. 1672-1678, doi: 10.1109/IVS.2018.8500658.

[31] S. Rezwana and N. Lownes, "Interactions and Behaviors of Pedestrians with Autonomous Vehicles: A Synthesis," *Future Transportation*, vol. 4, pp. 722–745, 2024. doi: 10.3390/futuretransp4030034.

[32] S. A. Bagloee, M. Tavana, M. Asadi, and T. Oliver, "Autonomous vehicles: challenges, opportunities, and future implications for transportation policies," *Journal of Modern Transportation*, vol. 24, pp. 284–303, 2016. doi: 10.1007/s40534-016-0117-3.

[33] S. A. Yusuf, A. Khan, and R. Souissi, "Vehicle-to-everything (V2X) in the autonomous vehicles domain – A technical review of communication, sensor, and AI technologies for road user safety," *Transportation Research Interdisciplinary Perspectives*, vol. 23, 2024. doi: 10.1016/j.trip.2023.100980.