

Deep Learning-based Trajectory Prediction for Autonomous Vehicles



By

Muhammad Nauman

(Registration No: 00000364573)

Department of Robotics & Artificial Intelligence

School of Mechanical and Manufacturing Engineering

National University of Sciences & Technology (NUST)

Islamabad, Pakistan

(2024)

Deep Learning-based Trajectory Prediction for Autonomous Vehicles



By

Muhammad Nauman

(Registration No: 00000364573)

A thesis submitted to the National University of Sciences and Technology, Islamabad,

in partial fulfillment of the requirements for the degree of

Master of Science in
Robotics and Intelligent Machine Engineering

Supervisor: Dr. Shahbaz Khan

Co-Supervisor: Dr. Muhammad Irfan

School of Mechanical and Manufacturing Engineering

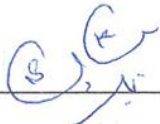
National University of Sciences & Technology (NUST)

Islamabad, Pakistan

(2024)

PROJECT / THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS Thesis written by Mr./Ms. Muhammad Nauman Registration No. 364573, of School of Mechanical and Manufacturing Engineering has been vetted by under signed, found complete in all respects as per NUST Statutes/Regulation, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS degree. It is further certified that necessary amendments as pointed by GEC members of the scholar have also been incorporated in the said thesis.

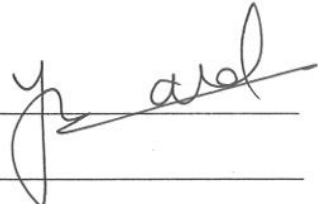
Signature: _____ 

Name of Supervisor: Dr. Shahbaz Khan

Date: 4/2/25

Signature (HoD): _____ 

Date: 6-2-2025

Signature (Dean/Principal): _____ 

Date: 6-2-25



National University of Sciences & Technology (NUST)

MASTER'S THESIS WORK

We hereby recommend that the dissertation prepared under our supervision by Muhammad Nauman (00000364573) Titled **Deep Learning-based Trajectory Prediction for Autonomous Vehicles** be accepted in partial fulfillment of the requirements for the award of MS in Robotics & Intelligent Machine Engineering degree.

Examination Committee Members

1. Name: Muhammad Irfan Zafar

Signature:

2. Name: Muhammad Tauseef Nasir

Signature:

3. Name: Muhammad Saqib Nazir

Signature:

Supervisor: Shahbaz Khan

Signature:

Date: 14 - Nov - 2024

14 - Nov - 2024

Head of Department

Date

COUNTERSIGNED

14 - Nov - 2024

Date

Dean/Principal

CERTIFICATE OF APPROVAL

This is to certify that the research work presented in this thesis, entitled "Deep Learning-based Trajectory Prediction for Autonomous Vehicles," was conducted by Mr. **Muhammad Nauman** under the supervision of Dr. **Shahbaz Khan**.

No part of this thesis has been submitted anywhere else for any other degree. This thesis is submitted to the **Department of Robotics & Artificial Intelligence** in partial fulfilment of the requirements for the Master of Science in the Field of **Robotics and Intelligent Machine Engineering** at the **School of Mechanical and Manufacturing Engineering, National University of Sciences and Technology, Islamabad**.

Student Name: Muhammad Nauman

Signature: 

Examination Committee:

a) External Examiner 1:

Signature:

.....

b) External Examiner 2:

Signature:

.....

Supervisor Name: Dr. Shahbaz Khan

Signature: 

Name of Dean/HOD: Dr. Kunwar Faraz

Signature: 

Author's DECLARATION

I certify that this research work titled "*Deep Learning Based Trajectory Prediction for Autonomous Vehicles*" is my work and has not been submitted previously by me for taking any degree from the National University of Sciences and Technology, Islamabad, or anywhere else in the country/ world.

If my statement is found to be incorrect at any time, even after I graduate, the university has the right to withdraw my MS degree.



Signature of Student

Muhammad Nauman

2021-NUST-MS-R&AI-364573

PLAGIARISM UNDERTAKING

I solemnly declare that the research work presented in the thesis titled “**Deep Learning-based Trajectory Prediction for Autonomous Vehicles**” is solely my research work with no significant contribution from any other person. Small contribution/ help, wherever taken, has been duly acknowledged, and I have written that complete thesis.

I understand the zero-tolerance policy of the HEC and the National University of Sciences and Technology (NUST), Islamabad, towards plagiarism. Therefore, I, as the author of the above-titled thesis, declare that no portion of my thesis has been plagiarized and that any material used as a reference is properly referred to/cited.

I undertake that if I am found guilty of any formal plagiarism in the above-titled thesis, even after the award of the MS degree, the University reserves the right to withdraw/revoke my MS degree and that HEC and NUST, Islamabad has the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized thesis.

Student Signature: _____



Name: Muhammad Nauman

DEDICATION

Dedicated to my exceptional parents and adored siblings whose tremendous support and cooperation led me to this wonderful accomplishment.

ACKNOWLEDGEMENTS

In the infinite grace and wisdom of Allah Subhana-Watala, I find my strength and guidance. It is with His divine light that I have been blessed and guided throughout this academic Endeavor and in all facets of my life. To Him, I owe all gratitude, for it is through His will that I have been fortunate to be surrounded by people who have been pillars of support and guidance.

My deepest appreciation goes to my parents who have been an ocean of love, resilience, and support. Their unconditional faith in me and prayers have been the bedrock on which I have built my aspirations.

To my dear siblings, who have walked with me through thick and thin, your belief in me and your constant encouragement has been invaluable. I am also profoundly grateful to my supervisor, Dr. Shahbaz Khan, who has been a mentor for excellence. His expertise, insights, and dedication have significantly shaped this work. Throughout this thesis, he provided invaluable guidance, constructive criticism, and a reservoir of patience. His unwavering support and wisdom have been pivotal in turning my aspirations into reality.

I would also like to extend my heartfelt thanks to my previous supervisor, Dr. Muhammad Jawad Khan, who taught me the foundational skills of conducting research, and to my CoSupervisor, Dr. Muhammad Irfan, whose valuable insights have guided me from the very start of this thesis. Their combined guidance has been instrumental in shaping the direction and quality of my research.

A special note of thanks to my friends who have been an enduring source of safety, laughter, and reassurance. Their constant presence and belief in my capabilities, especially during moments of self-doubt, have been immensely uplifting.

Lastly, everybody who has contributed, even in the slightest way, to my academic journey has my heartfelt gratitude. Whether it was a word of encouragement, a gesture of support, or a piece of advice, I cherish it deeply. To all, may Allah shower His countless blessings upon you, and may He guide us in all our endeavours.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	9
TABLE OF CONTENTS	10
LIST OF FIGURES	12
LIST OF TABLES	13
ABSTRACT	14
Chapter 1: Introduction	15
1.1 Why self-driving vehicles:.....	15
1.2 How self-driven vehicles are better than human-driven vehicles.....	16
1.3 Component.....	17
1.3.1 Sensors.....	18
1.3.2 Detection and Planning.....	18
1.3.3 Input.....	19
1.3.4 Control System	19
1.3.5 Decision-Making Module.....	20
1.3.6 Actuator	20
1.4 Motivation	20
1.5 Problem Statement.....	21
1.6 Objectives	21
• Minimum ADE: Our approach targets achieving minimal ADE.....	21
• Multimodality: We leverage a multimodal model for robust decisions.....	21
1.7 Proposed Solution.....	21
1.8 Thesis Organization	22
Chapter 2: Literature Review	23
2.1 Generative Adversarial Network-Based Approaches.....	23
2.2 Deep Learning-Based Approaches.....	24
2.3 Generative AI-based Approaches.....	24
2.4 Research Gap	25
Chapter 3: Proposed System.....	27
3.1 Dataset	27
3.2 Transformer Model	27
3.2.1 Attention Block.....	27
3.2.2 MLP Block	28

3.2.3 1D Convolutional Neural Networks (CNNs)	28
3.3 Proposed System Diagram.....	29
Chapter 4: Implementation	31
4.1 Dataset	31
4.2 Model.....	32
4.2.1 Input Block	32
4.2.2 Encoder Block	33
4.2.3 Fusion Block.....	37
4.2.4 Multi-Modality Block.....	38
4.3 Training Loss	40
Chapter 5: Results and Discussion	42
5.1 Multi-Modality Results.....	42
5.1.1 Comparison with Argoverse Baseline	42
5.1.2 Comparison with LaneGCN	43
5.1.3 Comparison with Lane Transformer	44
5.2 Results	45
5.2.1 Comparison with Argoverse Baseline	45
5.2.2 Comparison with LaneGCN	45
5.2.3 Comparison with Lane Transformer	46
5.3 Plots	48
Chapter 6: Conclusion and Future Work	49
6.1 Conclusion	49
6.2 Future Work	49
References	51

LIST OF FIGURES

Figure 1: Various Accident Pictures.	15
Figure 2: Road fatalities.	16
Figure 3: Autonomous vehicle components.	17
Figure 4: 1D CNN architectures.	18
Figure 5: Proposed model diagram.	19
Figure 6: Encoder Block.	34
Figure 7: Fusion Block.	37
Figure 8: Decoder Block.	39

LIST OF TABLES

Table 1: Dataset Detail.	31
Table 2: Comparison of LaneFormer with Argoverse.	40
Table 3: Comparison of LaneFormer with LaneGCN.	41
Table 4: Comparison of LaneFormer with Lane Transformer.	41
Table 5: Comparison of LaneFormer with Argoverse.	42
Table 6: parison of LaneFormer with LaneGCN.	43
Table 7: Comparison of LaneFormer with Lane Transformer.	44

ABSTRACT

Autonomous driving heavily relies on accurate trajectory prediction to optimize route planning and enhance vehicle safety. Current deep learning-based trajectory models have demonstrated remarkable success on public datasets but often fall short in real-time applications due to computational limitations in vehicles. In this research, we propose *LaneFormer*, an optimized trajectory prediction framework designed to balance high predictive accuracy with computational efficiency, ensuring its suitability for real-time deployment in autonomous systems. Our model introduces an efficient attention mechanism to capture complex interactions between agents and road structures, outperforming state-of-the-art methods while using fewer resources. We evaluate *LaneFormer* on the Argoverse dataset, demonstrating its robustness in predicting future trajectories with competitive metrics across multimodal scenarios.

Key Words: *Autonomous Vehicle, Transformer, Trajectory Prediction, Self-Attention, MultiModality, Argoverse Dataset.*

Chapter 1: Introduction

1.1 Why self-driving vehicles:

Self-driving cars are a relatively new technology with the ability to bring a drastic change in the way people move around. The manufacture and use of unmanned vehicles seek to respond to some of the greatest challenges, especially road safety, level of operation, and the negative influence on the surroundings. This advancement is achieved escorted by the scientific improvement of advanced sensors, machine learning algorithms, and even artificial intelligence that have been incorporated into self-driving cars, these technologies help in eliminating human error which causes a higher rate of road accidents. This safety improvement is significant as it speaks to the potential number of lives lost or aggravated injuries. Besides safety, self-driving cars minimize traffic and ensure better use of the roads. They are designable such that they are within a network that enables them to send alerts to each other and change course by the prevailing conditions on the roads, thus ensuring that the right routes are taken, there are little, or no gridlocks and time is saved. Also, through autonomous systems, constant speed safety and limits of the traffic rules can be controlled for better traffic flow and little to no violators of the rules. Self-driving cars are also expected to cut down on the pollutants emitted. That is, in driving patterns, speed, acceleration & braking, and even selection of the distance, autonomous cars are likely to use less fuel and energy more effectively. This optimization also goes in line with various global initiatives aimed at further minimizing the carbon consequences [1][2].

In addition, driverless cars have the potential to enhance the mobility of people unable to drive such as the aged and the disabled. Thus, providing a more accessible form of transportation. The implementation and use of autonomous vehicles in public transport and ride-hailing services can lead to more affordable and convenient transport solutions [2].



Figure 1: Vehicle accidents [3].

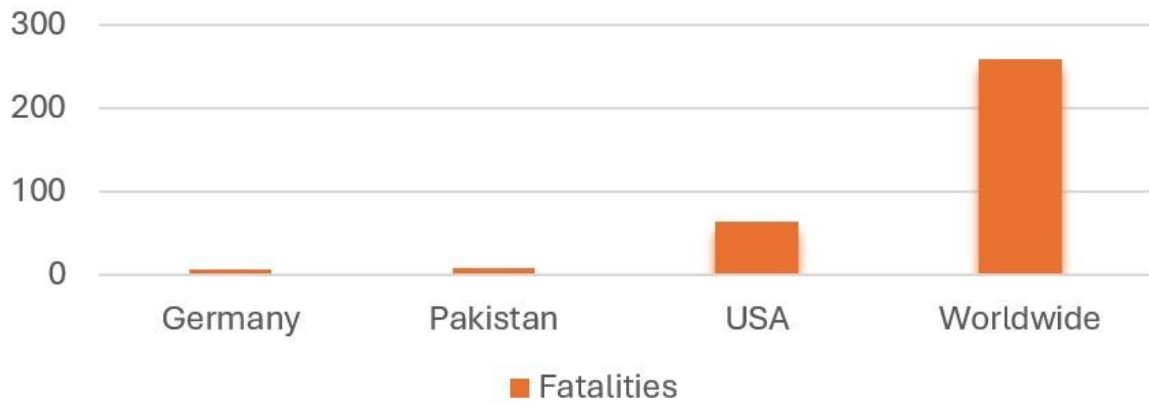


Figure 2: Road fatalities in 2021 – 2023 [3].

The main factors listed for these fatalities were speeding, intoxication, and distraction.

1.2 How self-driven vehicles are better than human-driven vehicles.

Self-driven vehicles, these vehicles provide a far better enhancement in road safety than human-driven vehicles. One of the major causes of this improvement is the fact that they do not involve human errors, which is the main cause of most of the traffic accidents. Pulling out from the statistics, distraction, tiredness, sober or compromised behavior, and emotional distress are some of the factors that affect how people drive. Autonomous vehicles, however, do not suffer such liabilities. These are equipped with complex computation algorithms together with an array of sensors and are limited by the lack of ex-devices that help process data that is being received in real-time [2][4].

Self-drive cars also employ several auxiliary and additional sensors like LIDAR, radar, various others, and cameras. This has the capability of tracking numerous objects and making reactions faster than a human driver. Besides, their models have prediction capabilities and algebraic functions telling them where any imagined person like pedestrians, cyclists, or vehicles would move, thus strategically avoiding any possible such collisions.

The cars are armed with a Vehicle-to-Everything (V2X) communication interface that allows communication with vehicles nearby, the surrounding traffic environment, and road conditions. These give enhanced awareness of the surrounding environment and the ability of the vehicle to predict and actively react to moving traffic situations more effectively than human drivers. Further, autonomous cars abide by each traffic regulation about speed and all other road rules always and under no circumstances creating room for risky behavior responsible for accidents and road rage. Their choices are unaffected by distractions, emotions, and other factors, which enhances safety on the roads and reduces unpredicted behavior [5][6].

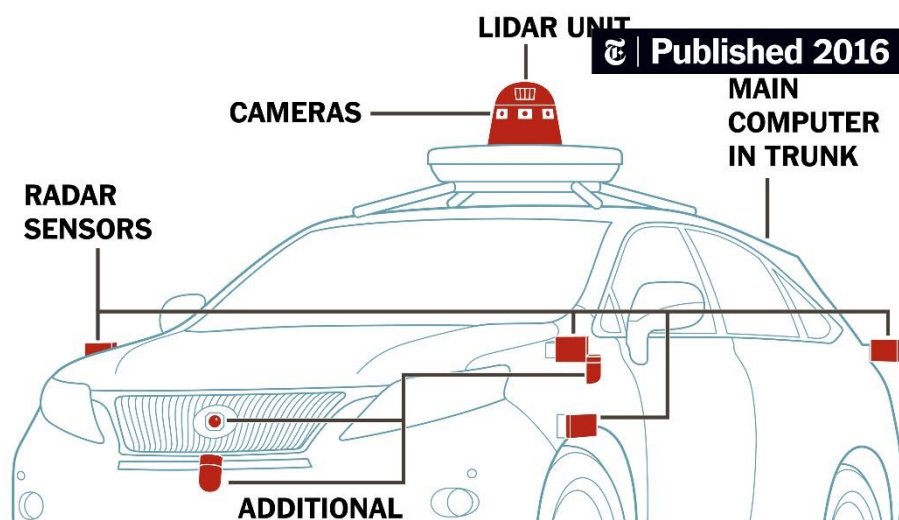


Figure 3: Components of Autonomous Vehicle [7].

1.3 Component

Autonomous vehicles, or self-driving cars, are used to be more advanced and efficient because they are made up of many different means of communication that work together to make navigation faster and safer. These components can be categorized in a more general manner into the following key areas [6][8].

1.3.1 Sensors

The sensors are the eyes and ears of an autonomous vehicle, receiving valuable and critical information about the surroundings. Information is obtained from different sources with the help of various types of sensors [9],

- **Lidar:**
Lidar systems use laser pulses and gauge the distance from the laser to the nearby surfaces. Collecting this enormous number of points helps the vehicle create a 3D image of all the surroundings to identify such things as obstacles, edges of the roadway, and a host of others [10].
- **Radar:**
Radar sensors detect objects using radio waves and measure their speed. These sensors are well utilized in harsh weather conditions such as fog and rain and are used in adaptive cruise control and collision avoidance functions.
- **Cameras:**
High-definition cameras record visual information in images and videos. They assist the vehicle in identifying lane markings, road signs, people, and other vehicles. Cameras play an important role in capturing information from the visual space essential for road movement.
- **Ultrasonic Sensors:**
Ultrasonic sensors are often used for short-range detection and aid in the low-speed operation of vehicles mostly during parking. These devices measure the distance of an object from the device by producing sound waves and calculating the duration taken for the echoes to return.

1.3.2 Detection and Planning

The perception system interprets the sensed data to recognize the environment around the vehicle. Several complex algorithms are involved in this process, such as computer vision and sensor fusion methods [11]:

- **Object Detection and Classification:**
The vehicle not only detects people inside but also identifies other objects outside of the car using advanced algorithms, including vehicles, pedestrians, traffic signs, and

obstacles on the road. This is done using images through Lidar cameras and video based on deep learning models built within the data.

- **Localization:**

Localization allows the car to position itself on a map properly. This component often incorporates GPS data and other information collected from sensors, such as Lidar or cameras, to allow the vehicle to achieve a high level of location accuracy even when the GPS signal is poor.

1.3.3 Input

Autonomous vehicles use detailed, high-definition (HD) maps to navigate complex environments. These maps contain information about road geometry, lane markings, traffic signals, speed limits, and more [12]:

- **HD Maps:**

HD maps are essential in guiding self-driving cars through busier areas. These maps show details related to the geometry of the roads, lane marks, traffic lights, and speed limits.

- **Path Planning:**

Using the provided HD maps that have been worked on previously, the vehicle can use the real-time generated data from the sensors in conjunction with the stationary environment for localization and path planning.

- **Global Planning:**

Uses map data to access the total end-to-end-point course relevant to the problem.

- **Local Planning:**

Adjust the vehicle's trajectory in real-time by avoiding obstacles and following traffic rules.

1.3.4 Control System

The control system follows the plan by sending the corresponding commands to the actuators of the different systems that are present in the vehicle like the throttle, brakes, steering, etc [9]:

- **Longitudinal Control:**

Regulates the motion of the vehicle through its speed by giving acceleration as well as applying the brakes on the vehicle.

- **Lateral Control:**
utilization of steering in controlling the direction of motion of the vehicle along a specified path or within a specified target lane. More advanced control methods like Proportional-Integral-Derivative (PID) controllers or Model Predictive Control (MPC) are applied to increase the stability of driving.

1.3.5 Decision-Making Module

This module is regarded as the brain of the vehicle as it perceives the surroundings makes decisions regarding various aspects and edits the above manual to suit its requirements. It receives inputs such as perception mapping and planning too [13]:

- **Predict Trajectories:**
Foresee the movement of cars, pedestrians, and other surrounding objects which is very critical in preventing accidents and making recommended path choices.
- **Behavior Planning:**
This will involve coming up with decisions that involve activities including but not limited to changing lanes, turning, and stopping. In the process of studying the environment, the vehicle will evaluate several factors, such as traffic conditions, lights, and pedestrian walkways, for timely responses.

1.3.6 Actuator

The actuators are the components that are responsible for controlling the motion of the vehicle. They execute commands sent by the control system for steering, throttle, brakes, and transmission by turning the electronic signals into motions [9].

1.4 Motivation

Given the critical role of autonomous vehicles in revolutionizing transportation, this research is motivated by the importance of the need to develop road safety and traffic management. Regardless of trends, horrifying statistics regarding road accidents, and the related fatalities and economic costs call for much better approaches and reliable models for trajectory prediction.

Many of the existing techniques for trajectory prediction fail to consider the practicality of the road by for example making wrong decisions that prompt an accident. Today, being in the Internet of things evolution there comes an opportunity to improve tackle trajectory prediction with greater accuracy, efficiency, and in real-time. Treating these features with respect, autonomous vehicles can considerably prevent accidents occurrence, enhance the performance of route determination systems, and promote comfort and safety for all road users.

1.5 Problem Statement

In autonomous driving, accurately predicting the future trajectories of surrounding agents, such as vehicles, pedestrians, and cyclists, is essential for safe and efficient navigation in dynamic environments. This task is particularly challenging due to the variability in agent behavior, complex road geometries, and the presence of multimodal interactions at intersections and other high-traffic areas. Using the Argoverse dataset, which provides high-definition map data and agent trajectory information, this study aims to develop a robust trajectory prediction model that can reliably forecast the future positions of target agents over a given time horizon. By leveraging detailed past trajectories and HD map features, the objective is to improve the model's ability to predict multiple possible outcomes with high accuracy, enabling it to handle diverse scenarios and enhance decision-making in real-world autonomous driving applications.

1.6 Objectives

This research aims to:

- **Minimum ADE:** Our approach targets achieving minimal ADE.
- **Multimodality:** We leverage a multimodal model for robust decisions.

1.7 Proposed Solution

The solution proposed in this thesis involves a two-stage pipeline for detecting railway track defects using advanced AI techniques. The first stage employs a YOLOv8 segmentation model, fine-tuned on a dataset of railway track images, to identify and segment various track components.

The segmented images are then processed to extract the rail surface, which is converted to grayscale and used to pre-train a U-Net-based model on a dataset of normal rail surfaces. This model is further fine-tuned on a subset of the Railway Surface Defects Dataset (RSDDs) to enhance its ability to detect surface irregularities. The combination of these two stages into a unified detection system offers a scalable and efficient solution for railway track inspection.

1.8 Thesis Organization

The structure of this thesis is as follows:

- **Chapter 2: Literature Review**
Provides an overview of existing research on railway track defect detection, highlighting the strengths and limitations of current methods.
- **Chapter 3: Proposed System**
Details the design and development of the AI-based detection system, including dataset preparation, model training, and system architecture.
- **Chapter 4: Implementation**
Discusses the implementation of the proposed system, covering the tools and technologies used and the challenges encountered during development.
- **Chapter 5: Results and Discussion**
Present the results of the performance evaluation of the system, analyzing its effectiveness in detecting railway track defects across different scenarios.
- **Chapter 6: Conclusion and Future Work**
Summarizes the key findings of the research, discusses its implications for railway safety, and outlines possible directions for future research.

Chapter 2: Literature Review

As evident from research, trajectory prediction using advanced deep learning models, such as Generative Adversarial Networks (GANs), offers significant potential in accurately forecasting the future positions of agents in dynamic environments. However, studies have consistently highlighted the computational challenges posed by these models. The large number of parameters required to capture complex agent interactions and environmental details makes these models computationally intensive. For instance, models like LaneGCN and attention-based GANs have shown improved accuracy but come with a high computational cost, requiring substantial memory and processing power. These challenges are particularly evident in real-time applications, where latency and resource constraints are critical. As research progresses, the focus is shifting toward optimizing these models to maintain high accuracy while reducing computational demand, paving the way for more practical deployment in scenarios like autonomous driving and robotics.

2.1 Generative Adversarial Network-Based Approaches

The theory of Generative Adversarial Networks (GANs) is quite popular, especially for its application in trajectory prediction for autonomous vehicles. The research reported here models the future movement of vehicles in dense traffic conditions to anticipate accidents. GAN-based models consist of two components: a generator that constructs probable future trajectories and a discriminator that ensures diversity in the generated trajectories. This framework shifts the classic artificial intelligence dilemma of distinguishing between real and fake to producing more realistic and believable future trajectories. The "upset vigorously signified" allows the convergence of the systems toward accurate predictions (et al., 2022).

The Adversarial training framework based on GANs handles the ambiguity and multiple modes of real traffic, acknowledging that each vehicle can have various possible target centres. GAN models enhance the predictability and stochasticity of trajectories, especially in complex road scenarios. Additionally, they help in visualizing dependencies between vehicle systems and road infrastructure, ensuring realistic behavior in crowded or complex traffic environments (et al., 2023). Equipped with driving scenario databases, models using GAN methods for trajectory prediction can perform well in diverse road and traffic conditions. GANs are increasingly used to predict challenges faced by self-driving cars. Although the network designs are

computationally expensive, including GANs in the trajectory prediction process improves the sensibility of autonomous system plans and navigation, making vehicles safer and more efficient [14], [15], [16].

2.2 Deep Learning-Based Approaches

Predictive modeling research in autonomous driving has neglected other types of modeling and embraced a deep learning approach because those models effectively handle agent-agent and agent-environment interactions. The main features of classical methods focus only on segmenting images and video frames to construct a scene. On the other hand, it is also worth mentioning that deep learning technologies based on Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), and Graph Neural Networks (GNNs) help predict trajectories more efficiently as they learn spatiotemporal patterns from the data (Bhattacharyya et al., 2018; Chai et al., 2019). They allow long-term dependencies to be maintained as in RNN architectures, while CNN modules are included to read rasterized maps for a more complex driving scene. Furthermore, the history of RNNs has been developed productively by empowering them with attention and, eventually, transformer techniques to concentrate on important details of traffic for better modeling (Zhao et al., 2020).

Another recent trend is the incorporation of multimodal trajectory prediction where the models elaborate on multiple future possible paths to capture the degree of uncertainty of the driving context. VectorNet (Gao et al., 2020), and LaneGCN (Liang et al., 2020) extend the targeted state-of-the-art by formalizing road geometry and interactions with agents through graphs that enable the model to work with interrelation between multiple objects of the scene. In this fashion, the deep learning models also utilize vectorized and graph-based encodings to give a better picture of the driving scene to enhance trajectory predictions. Such multistage strategies have been benchmarked on the Argoverse dataset with favorable results, showcasing the strong impact that deep learning can have to guarantee a safe and consistent autonomous system [4], [5], [6], [14], [17], [18][19], [20].

2.3 Generative AI-based Approaches

The acceleration of recent developments in Generative AI (GenAI) and particularly transformer architectures has been realized especially in trajectory prediction in autonomous driving

applications. While transformers were originally designed for language-based tasks, they are more than able to predict and model long-term dependencies and complex relationships in sequential tasks. For example, mmTransformer (Liu et al., 2021) and other transformer-based models (Casas et al., 2020) in trajectory prediction optimally use self-attention, which drives each of the agents in a scene and learns the interactions between the cars and the environment. These models enable the prediction of the scene using both the previous trajectories and highdefinition (HD) maps or both maps developing a beautifully informative narrative. Since Transformers capture spatial and temporal dependencies in the surrounding environment, these models can obviate the other relevant modality prediction to driving and resolve some of the uncertainties that exist in real-world driving navigation [21].

Additionally, it is worth pointing out that achieving multiple plausible future outcomes can also be supported more efficiently when Generative AI approaches are integrated into Transformer models. For instance, by incorporating VectorNet with a decoder based on Transformers, it has been shown that predicting turning angles can be significantly improved by graph-based encodings of interactions (Gao et al., 2020). Such approaches, like Multi-Head Attention Networks (Liang et al. 2020), pull the focus to different actors and features of roads to make a better prediction of the trajectory. Also, these GenAI methods utilize the efficiency of Transformers to combine information coming from different modalities, such as road geometry and vehicle states, to produce high-fidelity and robust multicomponent trajectory forecasts [16][17][18][19]. Deployment of such models on datasets including Argoverse has shown that future motion forecasts can be generated making a great contribution towards achieving autonomous navigation [1], [11], [12], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31].

2.4 Research Gap

As regards the previous research, it is apparent that there has been significant development in trajectory prediction. However, certain gaps persist. One such issue is the existing unimodal nature of most of the current methods. This deficiency hampers the performance of the model which is not able to foresee many potential future events appropriately, thus making the model less effective and flexible in more complex situations. There is also considerable debate about the relatively high average displacement error (ADE). This is due to some existing models being

unable to achieve a low average when predicting future states of agents. Overcoming these gaps is very important in enhancing the development of trajectory prediction models which are more accurate and flexible autonomous driving applications.

Chapter 3: Proposed System

In this chapter, we will discuss the proposed system to detect different defects in the railway track system. We will also discuss how we gathered our dataset and the different architectures of deep learning models we used.

3.1 Dataset

In this study, the Argoverse dataset has been utilized to train and assess the performance of trajectory prediction models. This dataset contains multiple driving scenarios, which include interactions with other actors in simple and complex city environments with varied roads and movement patterns. It encompasses relevant information for training models, such as vehicle trajectories, lane status information, and light signal status. It also helps the model understand the temporal dynamics of traffic by including both increasing traffic and non-peak periods. Further details regarding the Argoverse dataset will be discussed in the forthcoming chapter [32]

3.2 Transformer Model

The area of trajectory prediction has progressed because it is effectively able to model complex interactions with moving objects and their environments. In this paper, we present transformer-based models that incorporate previous and planned road structure information in addition to patterns of past movements of vehicles for future trajectory prediction. These models employ attention-based time and spatial features-based methods that revolutionize trajectory prediction. With the use of transformers, trajectory prediction allows for the inclusion of variance in behavior so that scenario driving in the predictive components can be done [22][8], [24], [33], [34], [35], [36].

3.2.1 Attention Block

The enhancement of the self-attention mechanism is important for trajectory prediction. It allows the model to pay attention to different parts of an agent's motion and the motion environments. In trajectory prediction, self-attention is used to track dependencies across multiple time slices and agents, evaluating the strength of each dependence dynamically. This helps the model to understand better both short- and long-range interactions in the scene. By

considering inputs such as past trajectories and environmental context, the self-attention block creates a more accurate representation that considers nearby agents and road structures. This broader perspective enables the prediction to encompass a more comprehensive range of possibilities, as it considers the agent's history and the likely interactions, resulting in more precise and real-life future trajectory predictions.

3.2.2 MLP Block

Fusion and processing of the encoded data for trajectory prediction through a transformer model would require an MLP block to be included. This MLP block is utilized to incorporate attentional MLP to the given features from the attention mechanism of the transformer. It aids in encoding the historical context of the agents and their actions. Furthermore, the MLP further optimizes the transformer's ability to predict in which direction the agents should be steered to do appropriate movements by passing outputs in a sequence of varying functional layers. This block incorporates spatiotemporal information well without altering the structure of the predicted trajectories concerning agents' intentions and road geometry, making the trajectory prediction system better [37].

3.2.3 1D Convolutional Neural Networks (CNNs)

Using 1D CNN within transformer-based trajectory prediction models helps process sequential motion data more efficiently. The 1D CNN is well-suited for preserving temporal information as it can effectively capture the short temporal structure within an agent's trajectory. In this architecture, the 1D CNN acts as a subunit that preprocesses the raw trajectory inputs before the attention mechanisms implemented by the transformer for the cross direction. Instead of just reinforcing short-term memories as in models with only CNN, this model captures short-term details using the CNN without losing the understanding of the long-term effects using the transformer. As a result, the model's forecasting performance is improved, and its robustness in various realistic driving situations is enhanced.

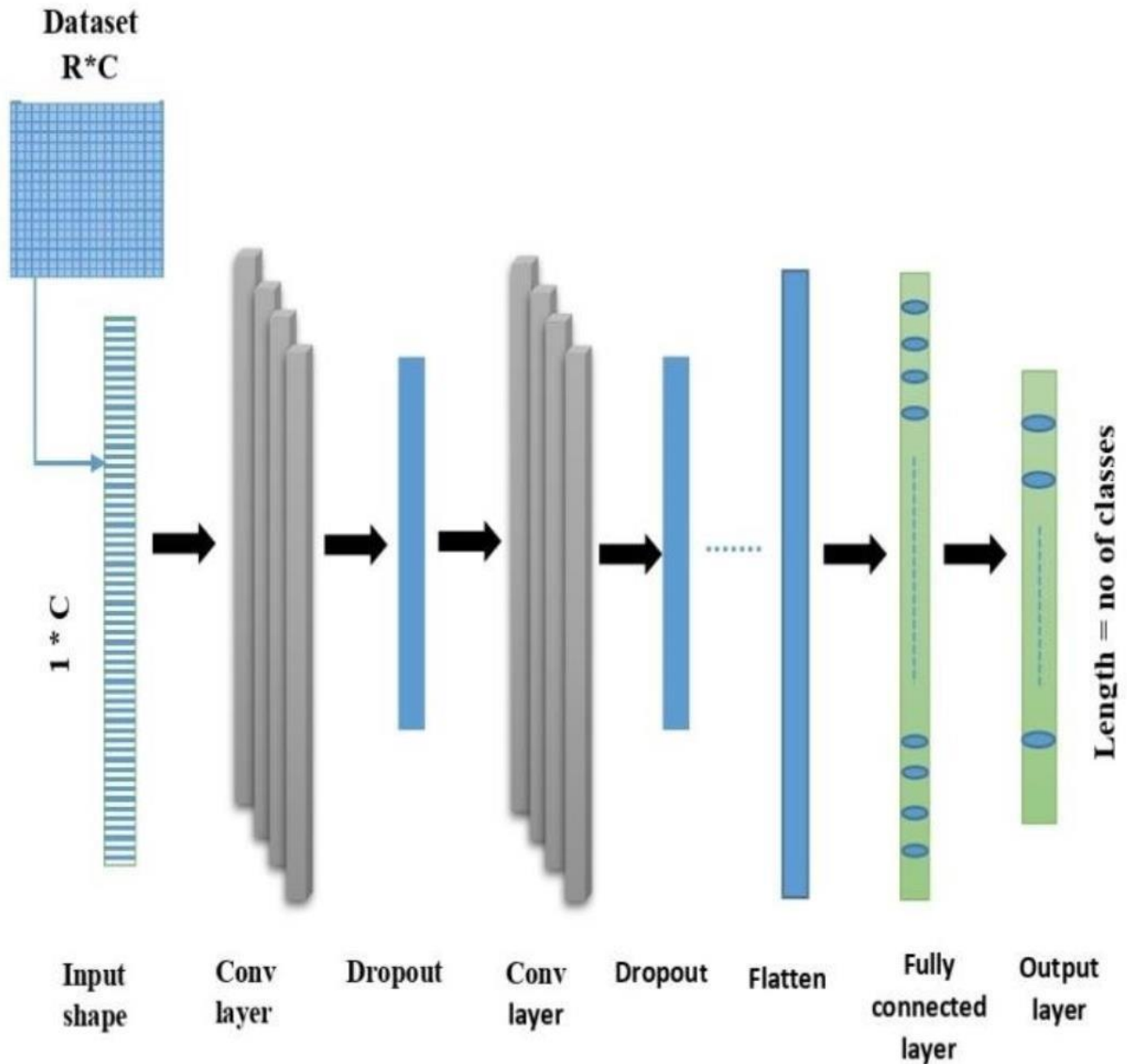


Figure 4: The simplest representation of 1D CNN Architecture (Amit et al, 2021).

3.3 Proposed System Diagram

This chapter deals with the author’s datasets – the data collected for training and its statistics the author addresses next. In other sections, the author addresses the different architectures that they employed while training and fine-tuning their datasets. To make it easier to follow all the processes, the schematic diagram of the proposed system is shown in Figure 11. In the next chapter, the author narrates how it was possible for them to integrate all these components, and therefore design their system.

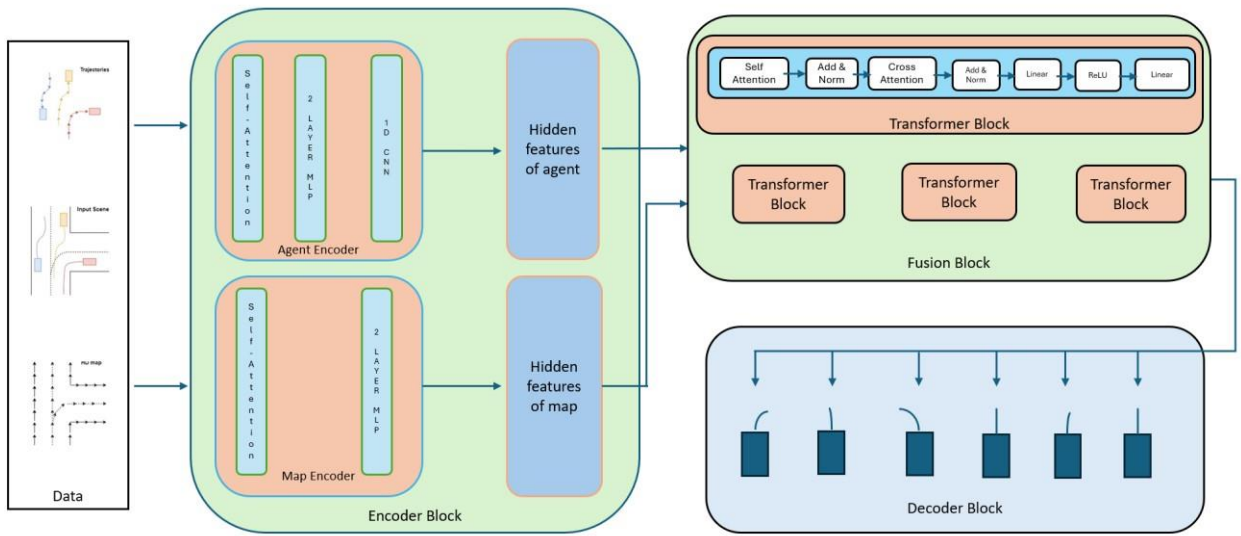


Figure 5: Proposed System Diagram.

Chapter 4: Implementation

This chapter provides an overview of the process of developing and implementing a trajectory prediction model based on the transformer architecture. It discusses how the proposed model handles complex traffic scenarios by considering both spatial and temporal dimensions in the interactions of various actors. The chapter also includes an evaluation of the new model in comparison to other trajectory prediction models to determine its efficiency and accuracy. Additionally, it explains the development and retraining of the model, focusing on attention-based feature extraction and feature fusion. Moreover, the chapter describes how the model incorporates multiple target behaviors and outlines steps taken to enhance prediction accuracy. Overall, this chapter aims to enhance understanding of the transformer model architecture used in this research and the methods employed to improve trajectory prediction accuracy.

4.1 Dataset

The Argoverse dataset contains detailed trajectories and high-definition maps with additional information such as speed limits and road characteristics. However, the dataset only provides location coordinates and lacks details about motion in space, such as angles and vertical falls. To address this issue, we introduce a vectorization operation to the roads and maps, converting them into variables that depict advanced geometry. When trails are recorded for agents, sampling is done at constant time intervals to create geometrically accurate points, which are then connected to form vector lines representing the path traversed by the agent. This technique helps prevent the loss of directional positions and aids the model in understanding movement dynamics. The same approach is applied to road sections, where the road is subdivided into equal portions known as splines or polylines. Constant distances are sampled on the roadmap, and adjacent points are joined to create vectors. A combination of these vector polygons explains the shape and orientation of individual road sections within the broader road network, making it more useful for the model [23][24].

Each vector is expressed as

$$d_i = [x_i , y_i , x_i^{pre} , y_i^{pre} , \Delta x , \Delta y , vid , sid]$$

represents the coordinates of the current point in a bird's-eye view $[x_i^{pre}, y_i^{pre}]$ Denotes its preceding neighbor's coordinates. The values Δx and Δy indicate the distance from the current point to the neighboring point on the current surface, defining the separation between roads or trajectories at a given moment. Additionally, the identifiers vid and sid represent the segment and vector of the scene, providing all the necessary information for proper construction. Converting trajectories and road segments into these vectors makes it easier for transformer models to understand local and global spatial relations more efficiently, thereby increasing the predictive power of the model.

Split	Number of Samples	Description
Training	205,942	Samples for model training
Validation	39,472	Samples for model validation
Testing	78,143	Samples for model testing

Table 1: Dataset split

4.2 Model

4.2.1 Input Block

The models that we have designed are built on the transformer framework and include four key blocks that will assist in solving the trajectory prediction problem. The first one is the input block, which aims to perform input data conversion of the input data such as HD maps and vehicle trajectories into vector form ready for any processing in the succeeding layers. This is important because the model requires this process since it encodes several details involving space and time directions into one simpler form. The input block provides a robust starting point

for precise and holistic trajectory prediction with input data from both the road structure and dynamic agents trajectories [17].

The next block is the encoder, which acts more like a purifier that employs the self-attention mechanism to pull out pertinent hidden features from the input data. Self-attention helps the model to focus on specific elements in a scene, including other vehicles, road shapes, or traffic signs and determines how important they are. In view of these interactions, the encoder can handle agents' and the environment's delicate but powerful relations. This helps to enhance the understanding of each agent dynamics and soil understanding of how agents interact in a traffic scene. The output of this encoder block is a set of vectors whose attributes are more sophisticated with respect to the agents and road features.

After the encoder comes the transformer blocks, which continue to generate the representation by combining the features of the agents and the roads and enhancing each of them. The blocks take advantage of the self-attentional mechanism to model complex interactions between the elements in the scene, determining how agents will behave with other cars and changes in the surroundings. The last element in the structure is a decoder based on multilayer perceptron MLP, which integrates the features that are obtained and accomplishes the prediction of the target agent's motion. In the MLP-based decoder, the information obtained in the previous blocks is put in use so that the trajectory is accurately predicted in terms of connecting short-range and long-range directional influences. With this structured and flexible architecture, the model adaptation in autonomous driving tasks makes it easier to cope with different cases in real life.

4.2.2 Encoder Block

1. Merging Trajectories and High-Definition Maps for Encoding:

After post-processing, the trajectories and the HD maps are encoded; that is, they are represented in vector form, which is the input to be gained from the performance model of the trajectory. These vectors feature in complex attributes like road patterns, moving vehicles and other activities around. They are structured in a way that makes the model learn the spatial relationships embedded in it. The feature vectors are then encoded through a structural phase transformation to achieve detail enhancement using a strong encoding mechanism. Based on earlier studies and empirical investigations, our model adopts an efficient and

performancebased encoder structure that incorporates compositional devices such as attention, CNNs and MLPs at the same time into the input.

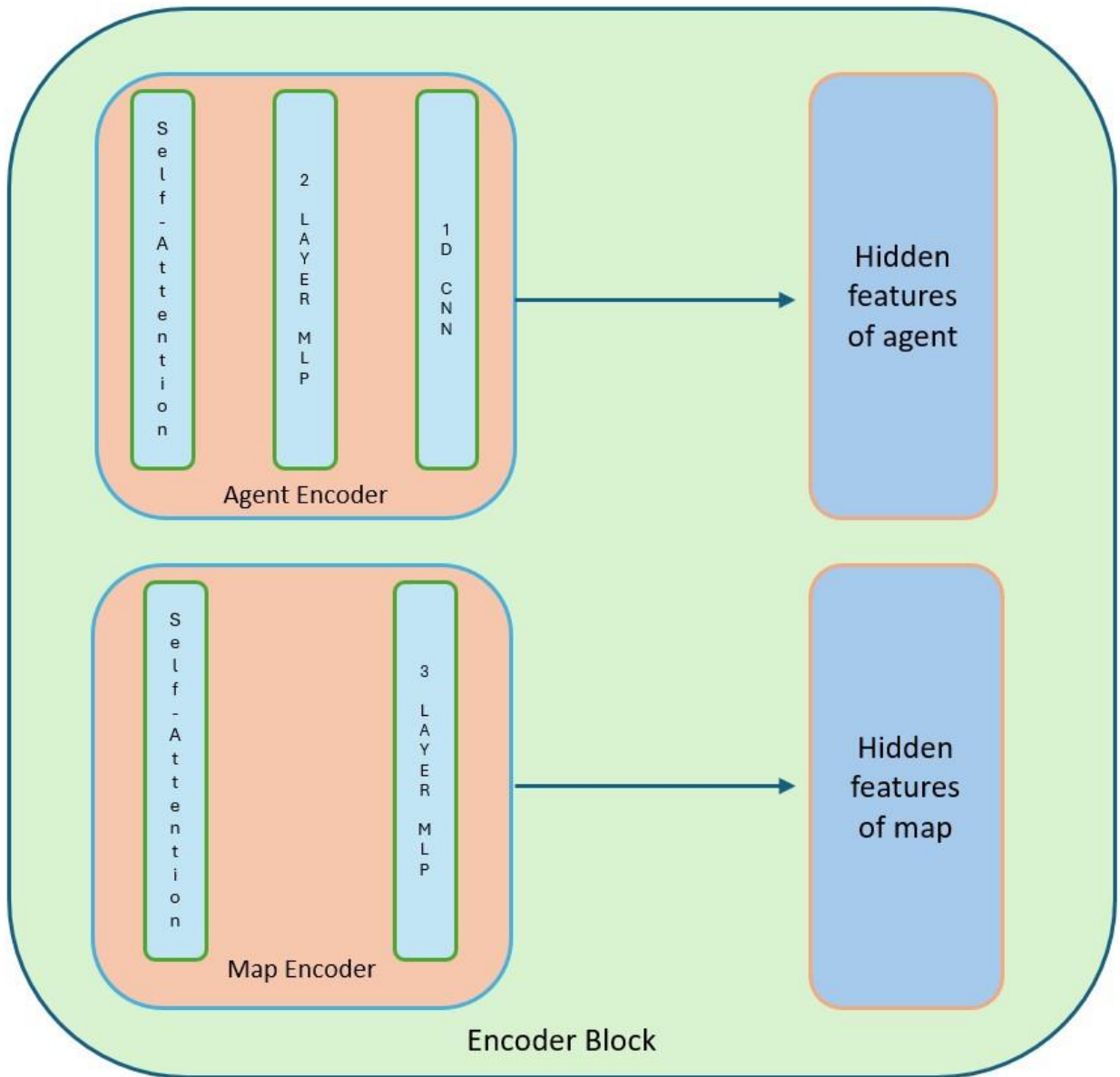


Figure 6. Encoder Block

2. MLP-Encoding of Road Information:

The encoder is based on a Multi-Layer Perceptron (MLP) and encodes road information. This component encodes input vectors corresponding to road segments at each time step. In the example above, the input vector fit refers to the characteristics of this road segment at this instant in time. This transformation is achieved using a three-layer perceptron with a "ReLU

activation function, denoted " ϕ ", and a learnable weight matrix W_{map} . The resulting feature matrix is structured in three dimensions: H, L, and M) where H signifies the number of hidden features, L indicates the fixed length of each segment (set to 10 meters for standardization), and M denotes the total number of road segments in the scene. This encoding allows the model to incorporate various road characteristics into a unified framework, preparing the data for subsequent processing [36].

3. Addressing Limitations of Individual Feature Vectors with Attention Mechanism: Using just the individual feature vectors is not enough to estimate the trajectory. For instance, while two road sections may look very similar in their beginning parts, differences may occur in their last parts that produce different geometric interpretations. This difference can be significant in how well the model can predict behaviors. To overcome this challenge, an attention mechanism is presented to successfully encode road segments into a set of feature vectors. The attention mechanism also assures the presence of inclusivity of both the local and global geometrical changes of the segment by letting the model attend to the key aspects of every segment. In this manner, each agent obtains a unique feature vector, which is used to determine the query, key, and value matrices needed in the self-attention mechanism.

4. Calculating Query, Key, and Value Matrices for Self-Attention:

The next step in the encoder is devoted to computing the query (q), key (k), and value (v) matrices. This task is basic to the self-attention mechanism, which enables the model to perform feature selection in the input data. The calculations can be formulated as follows:

$$q_{ti} = W^q f_{it}, k_{ti} = W^k f_{it}, v_{ti} = W^v f_{it},$$

where W^q , W^k and W^v are learnable weight matrices. These matrices are then passed through a weighting block utilizing the softmax function to normalize their contributions

$$h_{ti} = \text{softmax} (q \sqrt{d_k} \cdot k_{ti}^T) v_{ti}$$

Here, d_k represents the key matrix's length, ensuring that the model properly scales the attention scores. $h_i = \delta_{agg} (h_{ti}; W_{agg})$

This process enables the aggregation of features using a two-layer MLP, denoted as δ_{agg} , across road segments, resulting in a comprehensive feature matrix structured as (H, M).

5. Encoding and Aggregating Agent Trajectories:

A similar encoding scheme is employed in this step for the agents' trajectories. Each trajectory vector is operated on by an encoder block, creating a corresponding feature vector. For example, if two vehicles move together in the first half of their paths and obstruct each other's motion in the second half, their combined future course of movement will be dissimilar. To resolve this issue, we also utilize a self-attention block within the observation period to encode the whole trajectory to create one feature vector for each agent instead of time sequentially encoding portions of the trajectory at a time. This architecture uses self-attention, which helps keep each agent's temporal variations and dynamics in view and facilitates a better representation of each agent's motion.

6. Smoothing Trajectory Data with 1D CNN in Agent Encoder:

The agent encoder has the task of correcting the irregular nature of trajectory data, which can be caused by uncertain GPS tracking. This irregularity, also known as non-smoothness, can impact the accuracy of vehicle trajectories, posing a challenge for trajectory prediction models. This issue was addressed by incorporating a 1D CNN in the agent encoder. Unlike only an MLP, a 1D CNN can process a more significant portion of the trajectory due to its broader receptive field. The convolutional operation of the 1D CNN helps to reduce noise and mitigate the impact of insufficient data on trajectory prediction quality. By integrating the 1D CNN, the model can make accurate movement predictions even with noisy input data [38].

4.2.3 Fusion Block

The fusion process in transformer-based trajectory prediction models plays a vital role in combining the information of roads and the agents' interactions. So, in our setup, we prepare agents and road feature vectors into N by L and M by L 2D matrices, respectively. Where N is the number of agents, M is the number of roads, and L is the vector dimension of an agent or road segment. The future movement of a particular vehicle cannot only be based on its past movements but also consider the motions of surrounding agents and the road shape. Thus, low- and high-level interactions within the traffic scene are essential to forecast movements accurately.

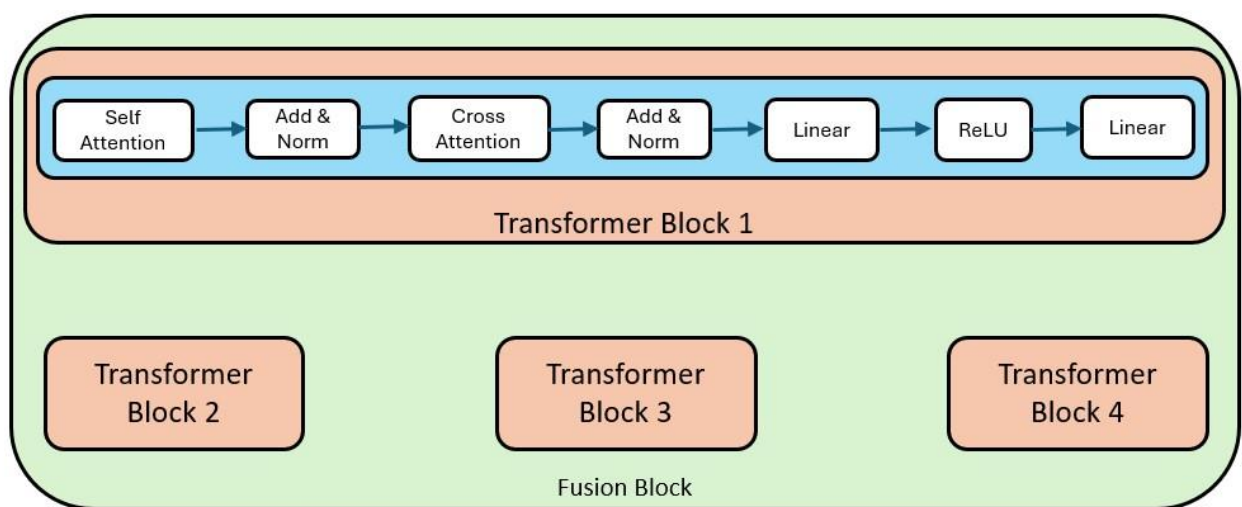


Figure 7. Fusion Block

For instance, when there are three vehicles, A, B, and C, A must understand the interaction with B and C and the geometry of the surroundings. At the same time, the model should combine these interactions to understand the model of the world or the so-called global interaction. To address the complex structure of local and global interactions, we utilize a hierarchical information fusion method employing the transformer architecture in this proposal. This hierarchical fusion approach incorporates local and global traffic scene information in the fusion process. The mechanism known as self-attention is of primary importance in this function, as it acts as the critical data-fusing node in the transformer architecture. Self-attention enables the machine learning model to emphasize specific regions of the input features, such as other agents and their environment, that the model perceives as applicable to the current task. When this hierarchical strategy is adopted, the model can maintain the deep structures of road and agent behaviors and improve its accuracy in prediction tasks [39].

For local information fusion, a transformer-style decoder is employed. In this structure, the agent and road matrices include a feature within which the multi-head attention block operates as a self-attention mechanism to focus on the more important features of the agent and road matrices. Another multi-head attention block also acts as an information collector, acting as a cross-attention layer that fuses information from the agent and road. A stack of four transformer layers is also utilized for additional performance improvements to be attained in capturing scenarios that are complex over the previous models. It is worth noting that the positions of road and agent features in this architecture can be switched, making it possible for the data to be integrated from the agent towards the road and the other way around. This facilitates the model by avoiding scenarios where users will be more prone to the features of one set of features. Rather, an equal assimilation of information can be achieved from both sets of features.

We apply the self-attention method to enable communication between all agents and all road segments by focusing on all interactions within the global context. Relying on this global fusion approach, the outcome produced is a feature vector that characterizes the target agent by containing all the spatial and interactional information concerning the environment surrounding the agent. The consideration of local and global interactions is done using different levels of hierarchy, and the model predicts the movements of the vehicles with optimal accuracy under different driving environments by understanding the details of every situation. This additional enhancement ensures that the model takes performance to the next level by determining with a high level of confidence which trajectories will most probably be taken.

4.2.4 Multi-Modality Block

The concept of multi-modality is essential in trajectory prediction for autonomous systems. It involves considering several potential future paths based on the agent's past movements and the current environment. For instance, a vehicle approaching an intersection could turn left, right, or continue straight. Consequently, the decoder of a trajectory prediction model must be able to predict more than one future path. This is done by setting a parameter k corresponding to the number of alternative trajectories the model can generate [40], [41].

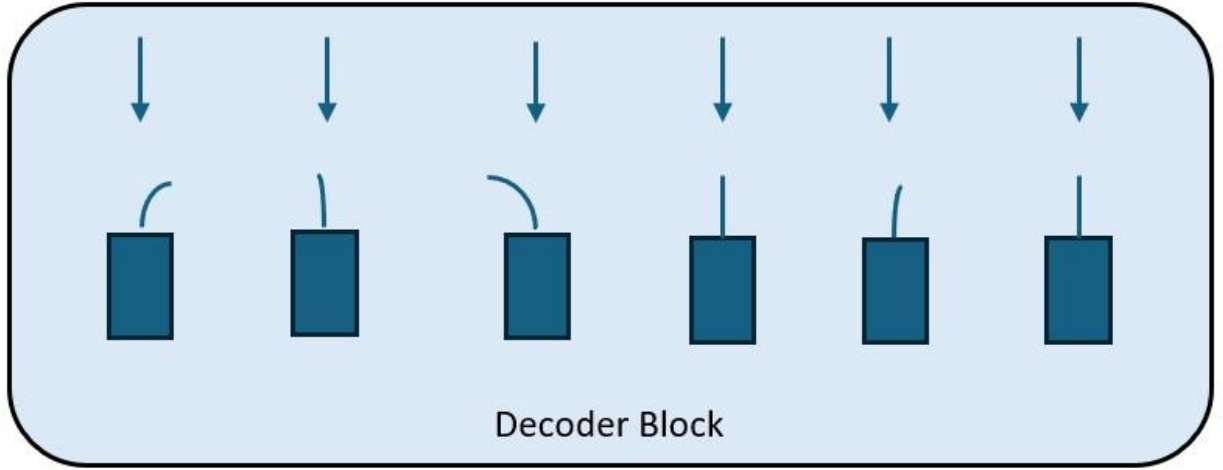


Figure 8. Decoder Block

Adopting multimodality is important to enhance the performance and robustness of vision systems, as it facilitates the model's emulation of the real world rather than focusing on one driving scenario.

The two basic contributions of this work are the same with respect to making the multi-modality more efficient using a multi-layer perceptron-based (MLP) decoder in the motion forecasting block. The approach involves embedding six trajectory decoders with the same design structure of a multi-layer perceptron (MLP). However, during the training phase, these six decoders behave differently as they are given distinct parameter configurations and will therefore be good for future trajectory modes. This ensures that the model can generate various, plausible outcomes due to various driving scenarios. The formulation

$$p_{k_i} = \Phi^{k_{dec}}(h_i; W_k) \quad \text{where}$$

p_{k_i} represents the k th predicted trajectory of the i th agent, depending on $\Phi^{k_{dec}}$, which is the k th trajectory decoder; h_i , a feature vector concerning the instantaneous agent i and W_k , a weight matrix for the k th decoder. This structure then enhances the model to understand the intricate details of various motion patterns, providing room for representing several possible movements.

$$c_i = \Phi_{scoring}(h_i; W_{scr})$$

The model also includes a scoring strategy for each predicted trajectory. A score is assigned to each trajectory based on its likelihood. This scoring mechanism is part of a decoding unit called

Φscoring, which inputs a weighted feature vector associated with each trajectory. This produces a vectored outcome that shows confidence in the chances of a trajectory being the final most probable one. Using dual trajectory decoders and confidence score systems, the model can forecast multiple possible scenarios while focusing on the most realistic trajectory to increase forecasting accuracy in a changing and turbulent environment .

4.3 Training Loss

The loss function used to train the trajectory prediction model based on the transformer consists of two main parts: the classification loss function and the regression loss function. The primary goal is to correctly classify the most probable predicted trajectory and accurately regress toward the actual trajectory. Multiple losses and models, such as LaneGCN, guide our implementation. The combined loss is the sum of these two individual losses [25][28][42].

$$\ell = \ell_{cls} + \ell_{reg}$$

This strategy ensures that the optimization procedures are effective for classification and regression tasks. The classification loss encourages the model to choose a specific trajectory, while the regression loss allows for the estimation of the chosen trajectory. To calculate the classification loss, the model selects the trajectory most accurately associated with the k-th ground truth at the last time step, referred to as k*. Once this best-associated trajectory is determined, it serves as a reference for further computations. The following notation applies to the classification loss:

$$\ell_{cls} = \frac{1}{N(K-1)} \sum_{i=1}^N \sum_{k \neq k^*} c_{ki} \max(0, c_{ki} - \epsilon - c_{ki^*})$$

Where c_{ki} indicates each confidence score of predicting trajectory k for the ith agent, and N is the total number of agents. The parameter ϵ equals 0.2 and is specifically a tolerance that has been put in place to ensure that the confidence score of the correct trajectory k* predicted is significantly higher than all the other predicted trajectories. This margin-based loss improves the model's performance by enabling it to focus on accurately distinguishing the positive class

trajectory. The regression loss component aims to match the predicted trajectory to the actual trajectory. It is computed as a weighted sum of two terms:

$$\ell_{reg} = \alpha \ell_{FDE} + (1 - \alpha) \ell_{ADE}$$

here, $\alpha \ell_{FDE}$ refers to the Final Displacement Error, which is the mean squared error MSE of the endpoint of the predicted trajectory k^* and the actual endpoint of the trajectory. The second term, ℓ_{ADE} is referred to as the Average Displacement Error, which in this case is taken as the standard Euclidean distance between k^* and the actual position over the whole time frame. The parameter α describes the effect of the weight applied to each loss component. While the model reduces the error at the final point, it also allows one to preserve the accuracy of the predicted motion over the entire sequence. In the end, these loss components are combined, leading to a more effective training process, resulting in better trajectories predictions.

Chapter 5: Results and Discussion

In this section, we present the performance of the two-stage trajectory predictor and how this performance varies when only one mode is used ($K=1$) or when multiple modes are allowed for inference $K=6$. The qualitative and quantitative evaluation focuses on minimum Average Displacement Error (minADE) and minimum Final Displacement Error (minFDE). The terms used here are aimed at providing a critical and rounded evaluation of the ability of the model to forecast trajectories over time. The minADE captures the typical variation between the predicted pathways and the movement for each time step. At the same time, the minFDE is an absolute measure at the last time frame and addresses the issue of how accurate the model gets while predicting the previous point of movement.

Our performance evaluation shows that in the multi-mode scenario $K=6$, the transformer model seems to learn the multiple models being estimated, which reduces minADE and min FDE values. It implies that the model is competent in understanding various motions and navigation uncertainties in the active driving system. We also see a similar pattern in the $K=1$ case; however, here, the model can predict the most likely path very accurately, but the model's total accuracy drops on the last position forecast. These results emphasize the models' capabilities

5.1 Multi-Modality Results

5.1.1 Comparison with Argoverse Baseline

Comparing the performance of LaneFormer with the Argoverse baseline model for multimodality $K=6$ reveals significant improvements across key metrics. LaneFormer achieves a minADE (minimum Average Displacement Error) of **0.80**, substantially lower than the Argoverse baseline's **1.71**, indicating a much closer alignment between the predicted and actual trajectories over time. Similarly, the minFDE (minimum Final Displacement Error) of LaneFormer is **1.21**, outperforming the baseline's **3.29**, showing that LaneFormer provides a more accurate endpoint prediction. Additionally, the Miss Rate of LaneFormer is **0.12**, markedly lower than the baseline's 0.54, which highlights its enhanced ability to predict the correct trajectory within a threshold, reflecting greater reliability and precision in diverse driving scenarios. These results demonstrate LaneFormer's superior accuracy and effectiveness in trajectory prediction.

Table 2: Comparison of LaneFormer with Argoverse.

Model (K=6)	minADE	minFDE	MR
Argoverse Baseline	1.71	3.29	0.54
LaneFormer	0.80	1.21	0.12

5.1.2 Comparison with LaneGCN

Compared to the works of LaneGCN, the LaneFormer model further enhances the existing work in terms of the net performance of the model on key trajectory prediction metrics for the multi-modality setting with **K=6**. It is interesting to judge how well LaneFormer outperforms LaneGCN in its minADE metric, as the authors report the lower minimum Average Displacement Error (minADE) amounting to **0.85**, whereas in LaneGCN, it is **0.87**. Likewise, this helps in lowering the minimum Final Displacement Error (FDE) that is noticed by all the participants of the competition, which is also cut down in LaneFormer, the minFDE measured at **1.31** against LaneGCN's **1.36**, emphasizing the better efficiency profile at the expected final position of all agents. Additionally, the miss rate of LaneFormer is said to be **0.14**, while that of LaneGCN is **0.16** means that there is better performance from LaneFormer in predicting realistic future trajectories. In a nutshell, these findings showcase LaneFormer as capable of predicting more realistic trajectories than other systems.

Table 3: Comparison of LaneFormer with LaneGCN.

Model (K=6)	minADE	minFDE	MR
--------------------	---------------	---------------	-----------

LaneGCN	0.87	1.36	0.16
LaneFormer	0.80	1.21	0.12

5.1.3 Comparison with Lane Transformer

The comparison of the performance of LaneFormer and Lane Transformer for the $K=6$ multimodality scenario shows that in prediction accuracy and prediction performance, LaneFormer is slightly better than Lane Transformer. The value of minADE (minimum Average Displacement Error) for LaneFormer equals **0.85**, which is also somewhat better than Lane Transformer's **0.86**, indicating that regarding the average trajectory prediction performance, Lane Former performs slightly better than Lane Constructor. Both models achieve the same value for minFDE (minimum Final Displacement Error), which is equal to **1.31**, meaning that their accuracy towards the end of the last predicted position is similar. Nevertheless, regarding the miss rate, LaneFormer scores **0.14** while Lane Transformer scores **0.15**, meaning that the likelihood of a true future path being missed slightly favors Lane Former. Such a result suggests that there is an overall improvement in how accurate multi-modal trajectory prediction .

Table 4: Comparison of LaneFormer with LaneGCN.

Model (K=6)	minADE	minFDE	MR
Lane Transformer	0.86	1.31	0.15
LaneFormer	0.80	1.21	0.12

5.2 Results

5.2.1 Comparison with Argoverse Baseline

When measuring the performance competency of LaneFormer with respect to the Argoverse baseline's $K=1$ model at the same level, improvements are evident in all key measures. LaneFormer obtains a minADE (Minimum Average Displacement Error) of **1.64** while the Argoverse baseline reaches **3.45** indicating how effective the relationship between actual and predicted trajectories is through time. Furthermore, the minFDE (Minimum Final Displacement Error) value for LaneFormer is **3.49**, which is lower than the baseline's **7.88**, thus meaning that LaneFormer enhances the prediction of the point where the trajectory will end. LaneFormer demonstrates a Miss Rate of **0.59**; this is considerably lower than the baseline, which reports a Miss Rate of **0.87**. While very few relationships regarding threshold values are considered, the greater effect on the predictive ability of the misclassification threshold rather than its actual value is evident. These results, taken into account, prove the reason why LaneFormer produces more accurate and effective trajectory predictions [32].

Table 5: Comparison of LaneFormer with Argoverse.

Model (K=1)	minADE	minFDE	MR
Argoverse Baseline	3.45	7.88	0.87
LaneFormer	1.64	3.49	0.59

5.2.2 Comparison with LaneGCN

Consider lane following tasks and trajectory prediction as an example. In particular, in the Lead vehicle level $K=1$ setting, the LaneFormer model is significantly more accurate than the LaneGCN in terms of Minimum Conversion metrics. For instance, LaneFormer records a minimum Average Displacement Error of **1.64** (minADE), a better achievement than that of

LaneGCN’s minADE of **1.71**, which places the agents’ average path forecasts under precision. LaneFormer is also noted to possess a minimum final displacement error of **3.49**, less than LaneGCN’s minimum final displacement error of **3.78**. The equivalent miss rates for LaneGCN are **0.59**, and for LaneFormer are **0.59**, which means that the Lane Former is more accurate than the latter in predicting reasonable trajectories of vehicles they might take in the future.

Table 6: Comparison of LaneFormer with LaneGCN.

Model (K=1)	minADE	minFDE	MR
LaneGCN	1.71	3.78	0.59
LaneFormer	1.64	3.49	0.59

5.2.3 Comparison with Lane Transformer

A comparative analysis of the accuracy and prediction of collision with respect to LaneFormer and Lane Transformer for the **K=1** scenario shows that LaneFormer performs somewhat better than Lane Transformer. For LaneFormer minADE calls, a **1.64** value was obtained. In comparison, Lane Transformer achieved a **1.75** value, which indicates that, on average, LaneFormer gives more correct trajectory predictions than Lane Transformer. The minimum final displacement error (minFDE) showed that all the models had the same value of **3.49**, suggesting that the models’ accuracy at the last predicted point was similar. Additionally, data on the percentage of misses indicates that only **0.59** of the recorded Miss rate for the new system was much better than the recorded **0.59** for Lane Transformer, suggesting that the new model missed future true trajectory a little less often than the comparator model .

Table 7: Comparison of LaneFormer with LaneGCN.

Model (K=1)	minADE	minFDE	MR
Lane Transformer	1.75	3.84	0.59
LaneFormer	1.64	3.49	0.59

5.3 Plots

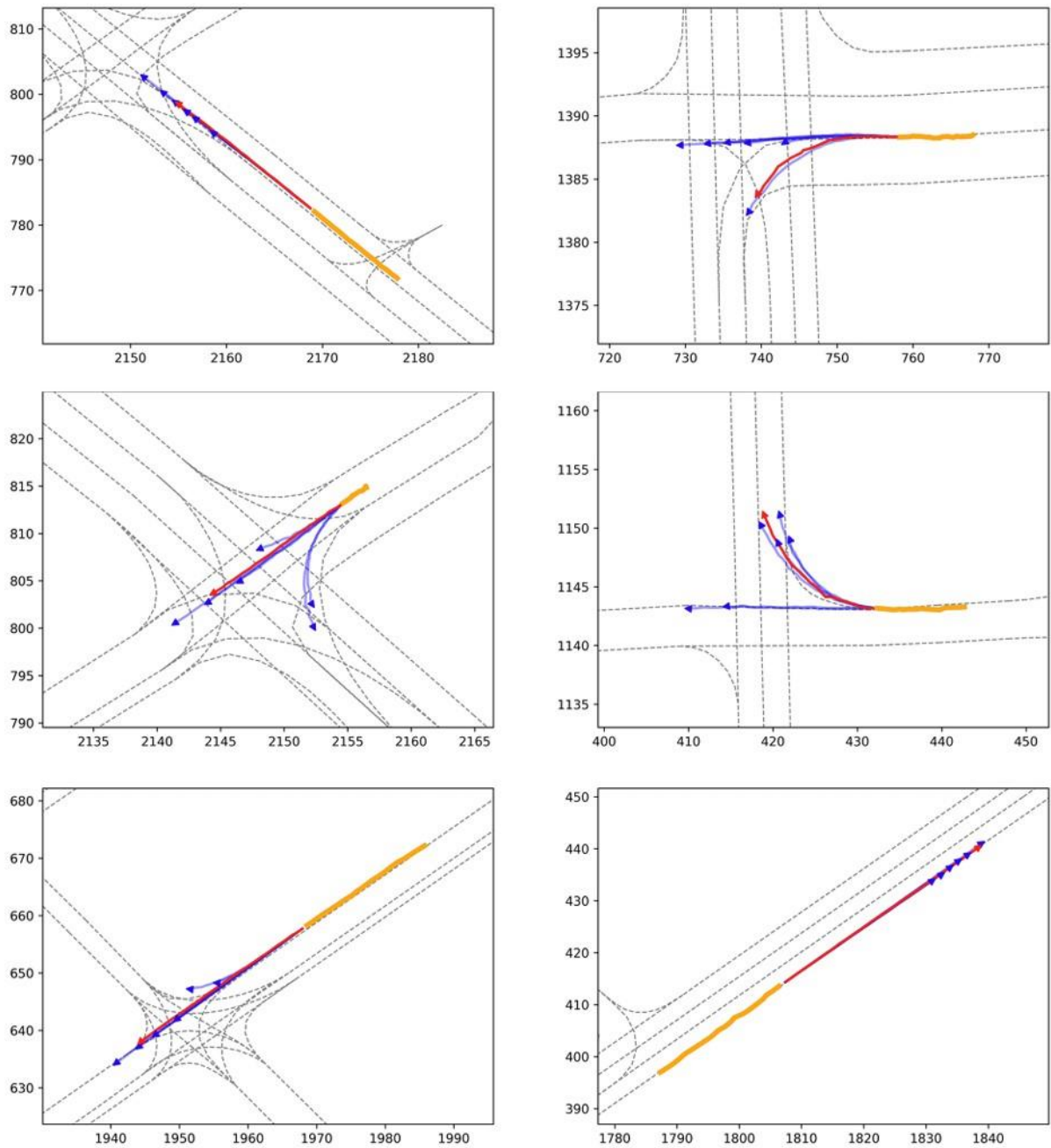


Figure 6: Visualization of the road and the trajectories of the target agent on the Argoverse validation set. The red, blue, and orange lines represent the ground truth, predicted trajectories, and observed trajectories, respectively.

Chapter 6: Conclusion and Future Work

This chapter summarizes the results achieved from the present study. It also discusses the proposed system's limitations and mentions the possible future work of this thesis.

6.1 Conclusion

The research proposes a motion forecasting system that can accurately predict the trajectory of autonomous vehicles in complicated driving conditions. In this research, we introduced the LaneFormer model, an optimized trajectory prediction framework designed to improve the predictive accuracy of trajectory forecasting. Through the integration of an attention-based mechanism and the refinement of a vectorized approach, LaneFormer successfully addresses the computational constraints that are often overlooked by conventional deep-learning models. Our approach demonstrates superior performance, particularly in multimodal trajectory prediction, where multiple possible outcomes are crucial for real-world scenarios. By effectively balancing the requirements of high accuracy with resource efficiency, LaneFormer outperforms existing models such as LaneGCN and LaneTransformer, particularly in key metrics like minADE, minFDE, and MR, showcasing its capability to handle complex dynamic environments while remaining computationally feasible for onboard vehicle systems.

The approach presented in this paper provides an upper-level, precise, and adaptable solution for the autonomous vehicle trajectory forecasting problem and is suitable for application in most autonomous systems across the globe.

6.2 Future Work

Future work on trajectory prediction using the Argoverse 2 dataset can focus on advancing multimodal prediction models that leverage the enhanced data quality and diversity of Argoverse 2 compared to Argoverse 1.1. Given Argoverse 2's more accurate representation of complex urban environments and its emphasis on capturing multiple plausible future trajectories, future research could explore designing models that better handle uncertainty and predict multiple potential outcomes for each agent. This could include developing improved multimodal architectures, such as advanced transformer-based or graphbased models, capable of learning diverse agent behaviors and interactions across various traffic scenarios. Additionally, optimizing these models for real-time inference, even with the added complexity of multimodality, will be essential for practical applications. Further research could also investigate transfer learning techniques that allow models trained on Argoverse 2 to generalize

effectively across other urban driving datasets, enhancing their robustness and applicability in different geographical locations and traffic conditions.

References

- [1] Y. Jeong, S. Kim, and K. Yi, “Surround vehicle motion prediction using lstm-rnn for motion planning of autonomous vehicles at multi-lane turn intersections,” *IEEE Open Journal of Intelligent Transportation Systems*, vol. 1, no. 1, pp. 2–14, 2020, doi: 10.1109/OJITS.2020.2965969.
- [2] X. Liu, Y. Wang, K. Jiang, Z. Zhou, K. Nam, and C. Yin, “Interactive Trajectory Prediction Using a Driving Risk Map-Integrated Deep Learning Method for Surrounding Vehicles on Highways,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 19076–19087, Oct. 2022, doi: 10.1109/TITS.2022.3160630.
- [3] Prof. Dr. Andreas Geiger, “Road fatalities in 2021-2023.”
- [4] K. Chen, X. Song, H. Yuan, and X. Ren, “Fully Convolutional Encoder-Decoder with an Attention Mechanism for Practical Pedestrian Trajectory Prediction,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 20046–20060, Nov. 2022, doi: 10.1109/TITS.2022.3170874.
- [5] M. Liang *et al.*, “Learning Lane Graph Representations for Motion Forecasting.”
- [6] J. Hong, B. Sapp, and J. Philbin, “Rules of the road: Predicting driving behavior with a convolutional model of semantic interactions,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Jun. 2019, pp. 8446–8454. doi: 10.1109/CVPR.2019.00865.
- [7] “Components of self-driving vchicles.”
- [8] K. Zhang, X. Feng, L. Wu, and Z. He, “Trajectory Prediction for Autonomous Driving Using Spatial-Temporal Graph Attention Transformer,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 22343–22353, Nov. 2022, doi: 10.1109/TITS.2022.3164450.
- [9] D. Yu, H. Lee, T. Kim, and S. H. Hwang, “Vehicle trajectory prediction with lane stream attention-based LSTMs and road geometry linearization,” *Sensors*, vol. 21, no. 23, Dec. 2021, doi: 10.3390/s21238152.

- [10] J. D. Choi and M. Y. Kim, “A sensor fusion system with thermal infrared camera and LiDAR for autonomous vehicles and deep learning based object detection,” *ICT Express*, vol. 9, no. 2, pp. 222–227, Apr. 2023, doi: 10.1016/j.ict.2021.12.016.
- [11] N. Nayakanti, R. Al-Rfou, A. Zhou, K. Goel, K. S. Refaat, and B. Sapp, “Wayformer: Motion Forecasting via Simple & Efficient Attention Networks.”
- [12] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi, “Spatio-Temporal Graph Transformer Networks for Pedestrian Trajectory Prediction.” [Online]. Available: <https://github.com/Majiker/STAR>
- [13] C. Guo *et al.*, “Query-Informed Multi-Agent Motion Prediction,” *Sensors*, vol. 24, no. 1, Jan. 2024, doi: 10.3390/s24010009.
- [14] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Dec. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.
- [15] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, “Social LSTM: Human trajectory prediction in crowded spaces,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Dec. 2016, pp. 961–971. doi: 10.1109/CVPR.2016.110.
- [16] Z. Pei, X. Qi, Y. Zhang, M. Ma, and Y. H. Yang, “Human trajectory prediction in crowded scene using social-affinity Long Short-Term Memory,” *Pattern Recognit*, vol. 93, pp. 273–282, Sep. 2019, doi: 10.1016/j.patcog.2019.04.025.
- [17] J. Gao *et al.*, “VectorNet: Encoding HD Maps and Agent Dynamics from Vectorized Representation.”
- [18] K. Simonyan and A. Zisserman, “VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION,” 2015. [Online]. Available: <http://www.robots.ox.ac.uk/>
- [19] Y. Fan, X. Liu, Y. Li, and S. Wang, “Look Before You Drive: Boosting Trajectory Forecasting via Imagining Future,” in *IEEE International Conference on Intelligent Robots and Systems*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 5551–5558. doi: 10.1109/IROS55552.2023.10341509.
- [20] O. Scheel, N. S. Nagaraja, L. Schwarz, N. Navab, and F. Tombari, “Recurrent Models for Lane Change Prediction and Situation Assessment,” *IEEE Transactions on Intelligent*

- Transportation Systems*, vol. 23, no. 10, pp. 17284–17300, Oct. 2022, doi: 10.1109/TITS.2022.3163353.
- [21] Y. Han, Q. Liu, H. Liu, B. Wang, Z. Zang, and H. Chen, “TP-FRL: An Efficient and Adaptive Trajectory Prediction Method Based on the Rule and Learning-Based Frameworks Fusion,” *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 2210–2222, Jan. 2024, doi: 10.1109/TIV.2023.3279825.
- [22] C. Feng *et al.*, “MacFormer: Map-Agent Coupled Transformer for Real-Time and Robust Trajectory Prediction,” *IEEE Robot Autom Lett*, vol. 8, no. 10, pp. 6795–6802, Oct. 2023, doi: 10.1109/LRA.2023.3311351.
- [23] J. Ngiam *et al.*, “SCENE TRANSFORMER: A UNIFIED ARCHITECTURE FOR PREDICTING MULTIPLE AGENT TRAJECTORIES.”
- [24] F. Giuliari, I. Hasan, M. Cristani, and F. Galasso, “Transformer Networks for Trajectory Forecasting.”
- [25] Y. Yuan, X. Weng, Y. Ou, and K. Kitani, “AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting.” [Online]. Available: <https://www.yeyuan.com/agentformer>
- [26] 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, Mar. 2021, doi: 10.1109/TIV.2020.2991952.
- [27] Z. Liu *et al.*, “Swin Transformer: Hierarchical Vision Transformer using Shifted Windows,” in *Proceedings of the IEEE International Conference on Computer Vision*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 9992–10002. doi: 10.1109/ICCV48922.2021.00986.
- [28] Z. N. Li, X. H. Huang, T. Mu, and J. Wang, “Attention-Based Lane Change and Crash Risk Prediction Model in Highways,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 22909–22922, Dec. 2022, doi: 10.1109/TITS.2022.3193682.
- [29] K. Zhang, L. Zhao, C. Dong, L. Wu, and L. Zheng, “AI-TP: Attention-Based InteractionAware Trajectory Prediction for Autonomous Driving,” *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 73–83, Jan. 2023, doi: 10.1109/TIV.2022.3155236.
- [30] H. Zhao *et al.*, “TNT: Target-driveN Trajectory Prediction.”
- [31] J. Gu, C. Sun, and H. Zhao, “DenseTNT: End-to-end Trajectory Prediction from Dense Goal Sets,” in *Proceedings of the IEEE International Conference on Computer Vision*,

- Institute of Electrical and Electronics Engineers Inc., 2021, pp. 15283–15292. doi: 10.1109/ICCV48922.2021.01502.
- [32] M.-F. Chang *et al.*, “Argoverse: 3D Tracking and Forecasting with Rich Maps.” [Online]. Available: www.argoverse.org.
- [33] A. Vaswani *et al.*, “Attention Is All You Need,” 2023.
- [34] Y. Tian, A. Carballo, R. Li, and K. Takeda, “RSG-GCN: Predicting Semantic Relationships in Urban Traffic Scene With Map Geometric Prior,” *IEEE Open Journal of Intelligent Transportation Systems*, vol. 4, pp. 244–260, 2023, doi: 10.1109/OJITS.2023.3260624.
- [35] L. (Luke *et al.*, “End-to-end Contextual Perception and Prediction with Interaction Transformer.”
- [36] A. Dosovitskiy *et al.*, “AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE.” [Online]. Available: <https://github.com/>
- [37] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, “GOHOME: Graph-Oriented Heatmap Output for future Motion Estimation.”
- [38] X. Jia *et al.*, “Towards Capturing the Temporal Dynamics for Trajectory Prediction: a Coarse-to-Fine Approach.”
- [39] Y. Zhu, D. Luan, and S. Shen, “BiFF: Bi-level Future Fusion with Polyline-based Coordinate for Interactive Trajectory Prediction.”
- [40] G. Aydemir, A. K. Akan, and F. Güney, “ADAPT: Efficient Multi-Agent Trajectory Prediction with Adaptation.”
- [41] Y. Lu, W. Wang, X. Hu, P. Xu, S. Zhou, and M. Cai, “Vehicle Trajectory Prediction in Connected Environments via Heterogeneous Context-Aware Graph Convolutional Networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8452–8464, Aug. 2023, doi: 10.1109/TITS.2022.3173944.
- [42] L. Anthony Thiede and P. Prabhanjan Brahma, “Analyzing the Variety Loss in the Context of Probabilistic Trajectory Prediction.”