Project Plan for Simplified Vehicle Trajectory Prediction Using Deep Learning

Project Description

This project aims to develop a simplified deep learning-based solution for predicting the trajectory of a single vehicle over a short term. Successful predictions are crucial for enhancing safety and efficiency in autonomous driving systems and advanced driver-assistance systems.

The project will concentrate on predicting the path of an individual vehicle, thereby simplifying the model's complexity, and focusing on extracting meaningful patterns from the vehicle's past movement. Also, the prediction horizon will be limited to a few seconds ahead. Such a constraint not only makes the problem more tractable and saves us computational cost but also aligns with the immediate decision-making needs of autonomous driving systems which should react shortly for safe navigation avoid possible obstacles.

Literature Survey

Key literature and algorithms for the Vehicle Trajectory Prediction problem include:

- RNN and LSTM for Trajectory Prediction: Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks have been extensively used due to their effectiveness in handling sequential data. These models can capture temporal dependencies in vehicle movement data to predict future trajectories.
- Encoder-Decoder Architectures: These models are well-suited for sequence-to-sequence predictions, making them a strong candidate for encoding historical trajectories and decoding them into future positions.

Possible Data for Training

Waymo Open Dataset: Known for its high-quality and extensive range of sensor data, the Waymo dataset includes precise trajectory information that will be instrumental in developing a robust model which is able to understand and predict vehicle movements accurately.

Timeline

- March 1st: Finalize the algorithm choice, focusing on RNN, LSTM and Encoder-Decoder models.
- March 1st - 10th: Begin model implementation with Python in TensorFlow or PyTorch, including data preprocessing and initial training.
- March 11th - 19th: Continue refining the model with further training and validation to improve prediction accuracy.
- March 20th: Finalize and submit the project report and code.