Vehicle Motion Prediction Using the Waymo Open Motion Dataset

Eden Wang[1], Emil Vardar[2], Lovisa Byman[2]

{ eyw, eevardar, byman }@stanford.edu
[1]Computer Science, [2]Electrical Engineering Stanford University

Stanford
Computer Science

Summary

Working with the Waymo Open Dataset, we leveraged a two-part model that was able to generate predictions on an object's position up to 8 seconds in the future provided with one second of context. We found that:

- Modular approaches show significant potential as **intuitive, tractable models** for motion prediction tasks (see: Experiments)
- Iterating on our two-part model allowed us to discover **pertinent features** unique to each task.
- LSTM structures can implicitly **model latent behavioral states** and are promising in motion prediction architecture..

Background

A significant challenge to the adoption of autonomous vehicles is **motion prediction**.

Our work leverages the **Waymo Open Dataset**, a collection of scenarios containing "critical scenarios" that demand modeling complex object interactions[1, 2].

The dataset contains **310,062 segments**, each with the states of all active objects (including vehicles, bikes, and pedestrians) along with roadgraph and roadsign features.

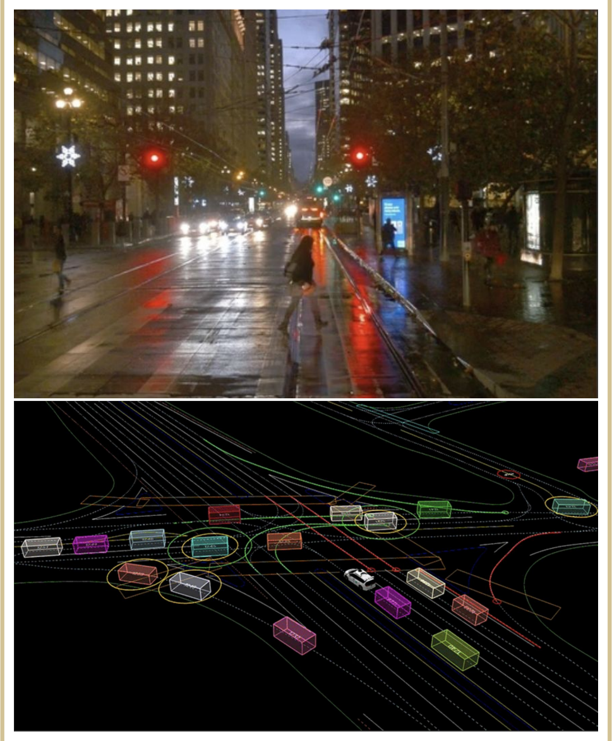

Fig. 1: Sample scenario data, including raw camera footage (top) & object modeling (bot)

One such scenario can be seen below:

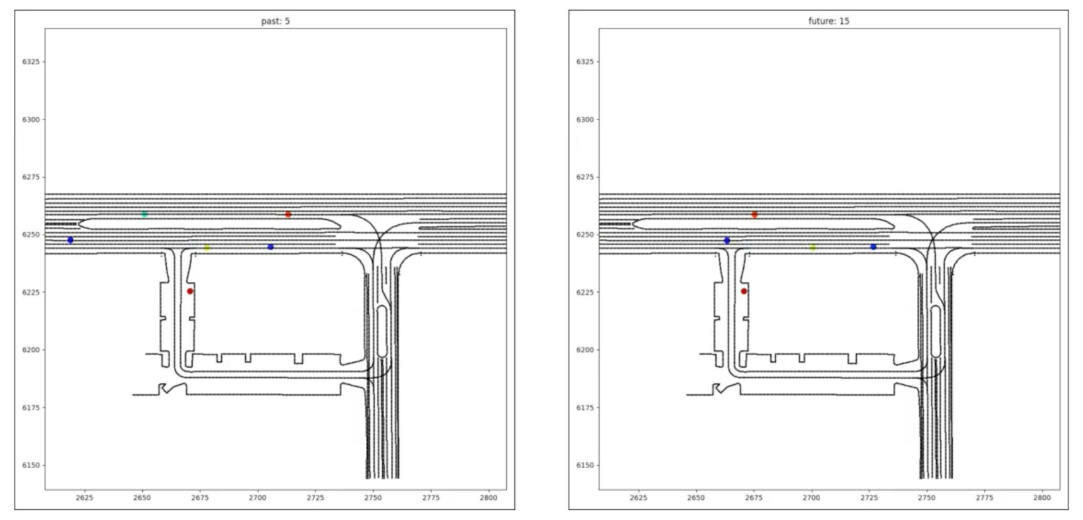

Fig. 2: A processed example scenario, displaying active agents and roadgraph features at timestamp 0 (left) and timestamp 20 (right).

Many approaches that have recently gained traction use **modular approaches with subtasks** rather than end-to-end models[3, 4]. Approaches also vary in feature extraction and pre-processing.

Consequently, the range of models used in the community varies greatly, ranging from simple end-to-end CNN models to complex composite models[5].

Technical Methods

Overview

We leverage a modular approach that first predicts object goals, then generates trajectories conditioned on these goals. Given the time dependencies present of object states, we rely heavily on **LSTM cells**, a recurrent structure that can retain information from cell to cell.

Both models use **mean squared error** as the loss function, as it naturally represents the distance from a ground truth. Both models are trained with an **Adam optimizer** ($\beta_1 = 0.9$, $\beta_2 = 0.999$). Additionally, we use two evaluation metrics: **average displacement error (ADE)** and **final displacement error (FDE)**.

Goal Prediction Module

This module predicts the endpoint of target objects 8 seconds in the future. Our model processes **time-dependent elements** (e.g., object velocity & yaw) and **static elements** (roadgraph) separately, applying convolutions to static features and LSTM cells to object states. Outputs are joined and used in a final dense layer that outputs predictions.

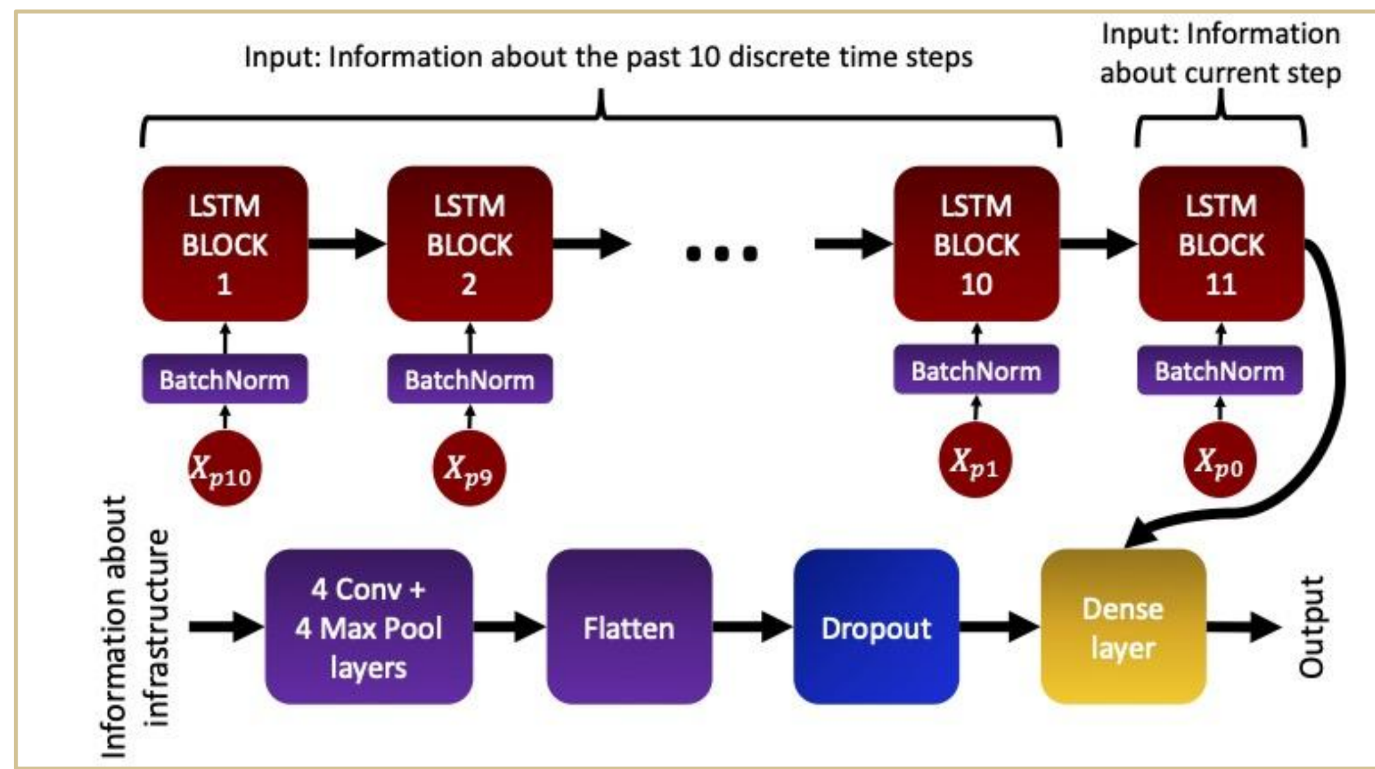
Input: Information about the past 10 discrete time steps
Input: Information about current step
LSTM BLOCK 1
LSTM BLOCK 2
LSTM BLOCK 10
LSTM BLOCK 11
BatchNorm
X_{p10}
X_{p9}
X_{p1}
X_{p0}
Information about infrastructure
4 Conv + 4 Max Pool layers
Flatten
Dropout
Dense layer
Output

Fig. 3: Model architecture for our goal prediction module. This architecture is also reused as the decoder for our trajectory completion module.

Trajectory Completion Module

This module predicts object trajectories over 8 second conditioned on an anchor endpoint. We leverage an **encoder-decoder structure**, with our encoder being similar to our goal prediction model (see **Fig. 3**) and our decoder consisting of **8 LSTM** cells that output trajectory predictions for each second in the future.

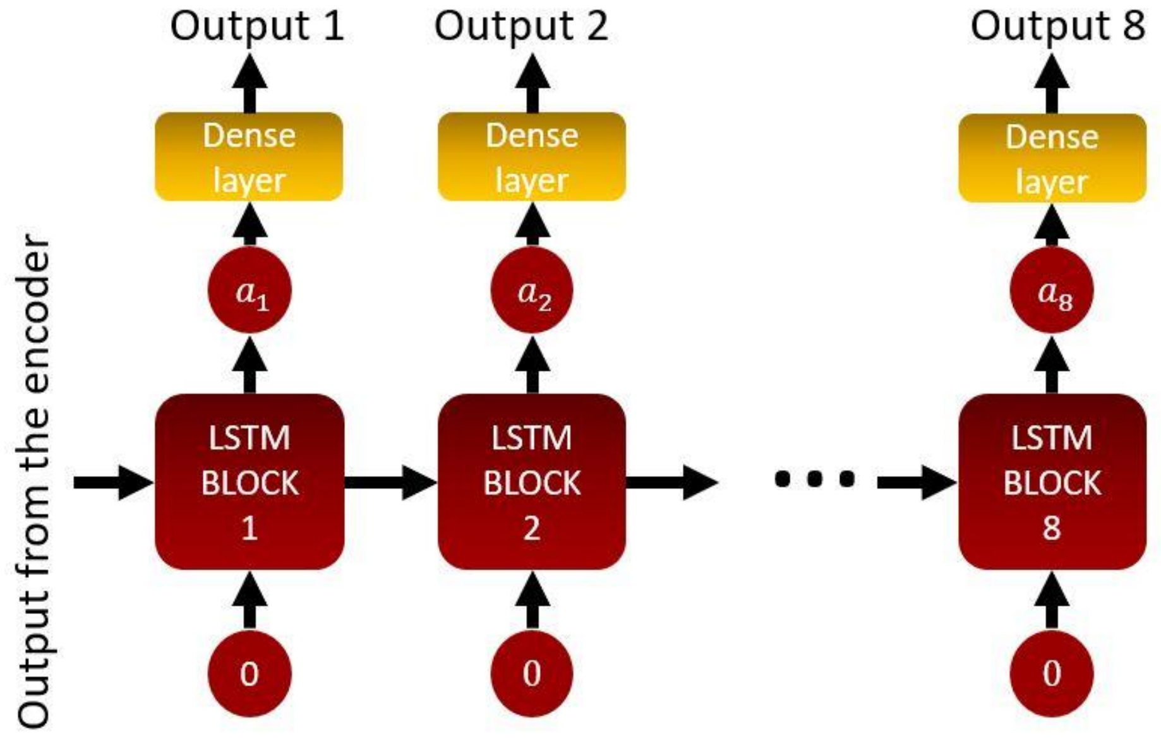

Fig. 4: Decoder architecture for our trajectory prediction module. Each LSTM cell is responsible for capturing behaviors between a given interval (0-1s, 1-2s, …, 7-8s).

Experiments

Goal Prediction

The best **validation loss** obtained by our goal prediction model was **523**, or a **distance of 22.8 meters**. The training loss was significantly smaller, so the low performance can be explained by overfitting. This can be prevented by increasing the regularization or by increasing the size of our dataset. The **minimum error** the model achieved on the **training set** was around **12.5 m**, which could be decreased even further.

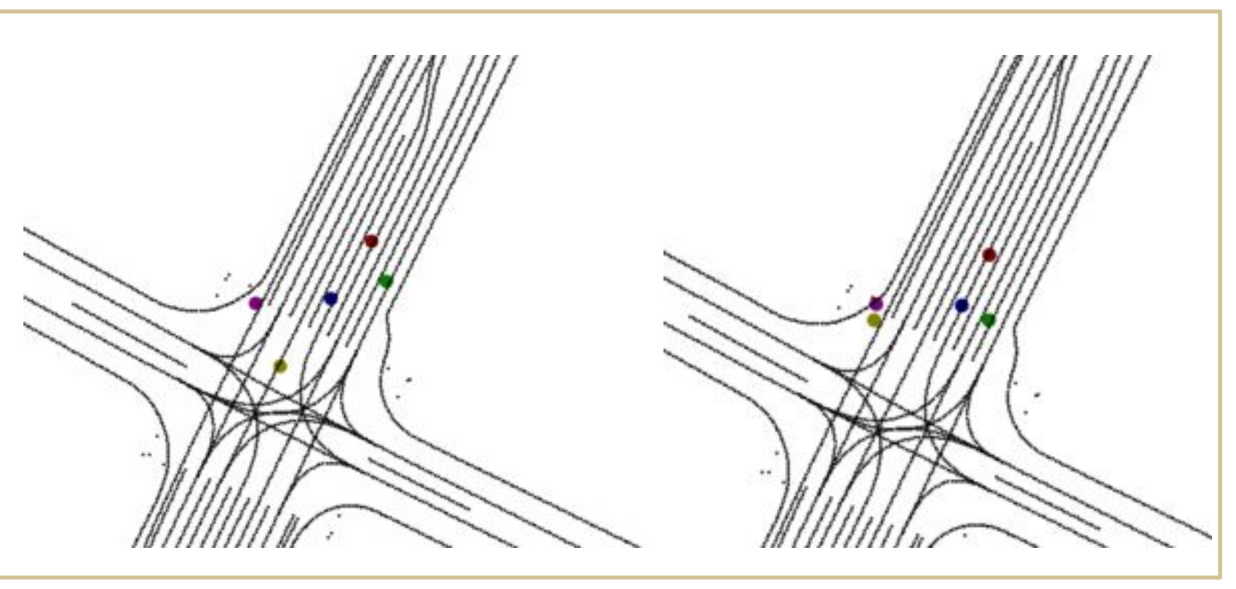

Fig. 5: The predicted goal positions (left) and the ground truth positions (right) for a **training example**. The predicted positions are very close to the ground truth positions.

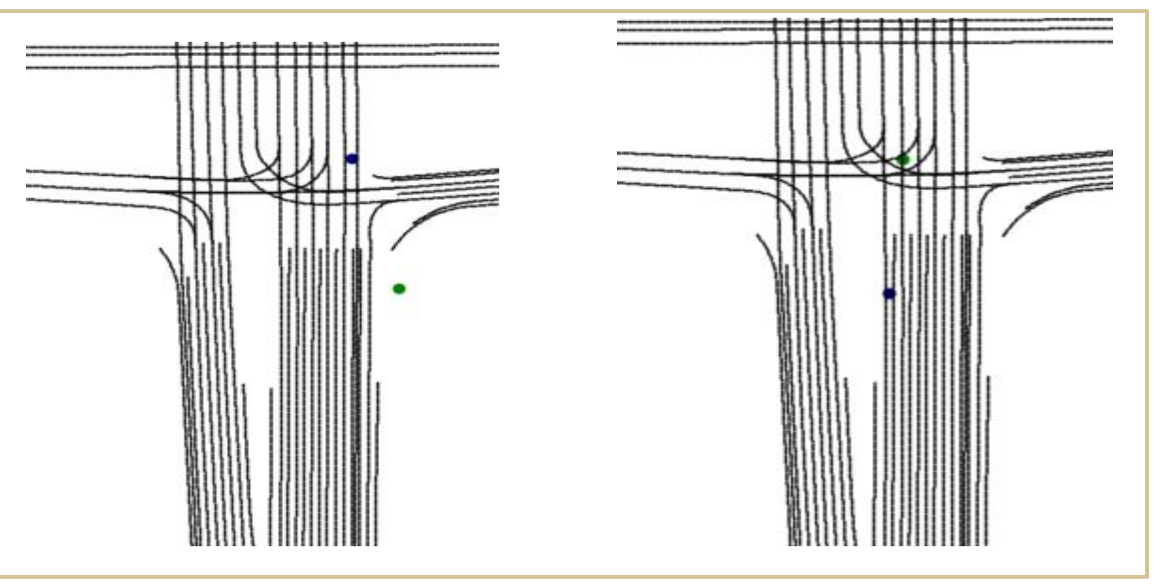

Fig. 6: The predicted goal positions (left) and the ground truth positions (right) for a **validation example**. The predicted positions are significantly further from ground truth positions.

Trajectory Prediction

Compared to the goal prediction model, both the training loss and the validation loss decreases much quicker. This is expected as it is an easier task to predict the trajectory with an anchor point compared to predicting end positions as far as 8 seconds into the future. The **ADE** was **8.0 m** and **7.6 m** for validation and training set respectively.

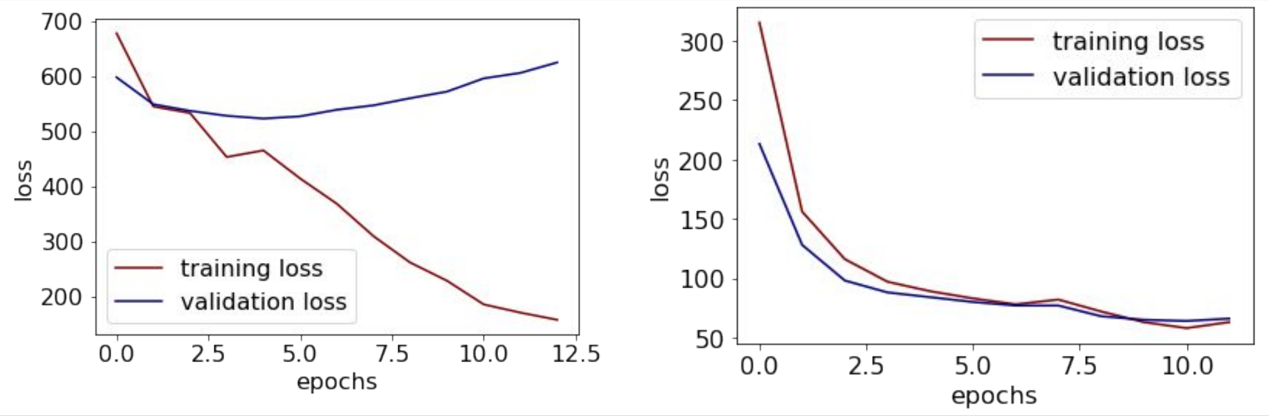

Fig. 7: Training and validation losses for our goal prediction (left) and trajectory prediction (right) modules.

Combined Model

The **ADE** for the combined model was **13.15 m** and the **FDE** was **23.95 m**.

References

[1] - Waymo. Waymo Open Dataset. URL: https://waymo.com/open/download/, 2022 [Online].

[2] - Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R Qi, Yin Zhou, et al. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 9710–9719, 2021.

[3] - Hang Zhao, Jiyang Gao, Tian Lan, Chen Sun, Benjamin Sapp, Balakrishnan Varadarajan, Yue Shen, Yi Shen, Yuning Chai, Cordelia Schmid, et al. Tnt: Target-driven trajectory prediction. arXiv preprint arXiv:2008.08294, 2020.

[4] - Junru Gu, Qiao Sun, and Hang Zhao. Densetnt: Waymo open dataset motion prediction challenge 1st place solution. arXiv preprint arXiv:2106.14160, 2021.

[5] - Stepan Konev, Kirill Brodt, and Artsiom Sanakoyeu. Motioncnn: A strong baseline for motion prediction in autonomous driving. In Workshop on Autonomous Driving, CVPR, 2021.