PREDICTING VEHICLE TRAFFIC BEHAVIOR WITH TIME SERIES MODELING - TRANSFORMERS VS LSTM

Ayush Srivastava
ETIM
Carnegie Mellon University
Pittsburgh, PA 15213
ayushsri@andrew.cmu.edu

Prakruthi Pradeep
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
prakruth@andrew.cmu.edu

Arnav Menon
Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
arnavmen@andrew.cmu.edu

May 4, 2024

ABSTRACT

The general objective of this research is to model Human Traffic Behaviour. The models developed in this paper will lay ground work for autonomous vehicles in taking decisions to ensure they meets their goals while ensuring collaborative safety. Specifically, we will look to model the velocity profile of vehicles at intersections and on highways. The approach discussed in this paper involves utilizing object detection and trackers like YOLO to extract trajectories of moving objects as well as Transformer and LSTM models to model the trajectory data's sequential and time series nature.

***K*eywords** Transformers · LSTM · Object Tracking · YOLO · Autonomous Vehicle · NGSIM

1 Introduction and Goals

Trajectory prediction is vital for autonomous vehicles to meet their goals while maintaining necessary safety conditions. Autonomous vehicles need to anticipate the future trajectories of other vehicles in their surroundings, in order to plan safe and optimal paths, and make decisions that align with traffic requirements. To this end, the goal of our project is to perform time series analysis on the NGSIM dataset to model vehicle behavior.

Our project implemented the following steps to meet this aim. Firstly, we were interested in utilizing Computer Vision techniques including YOLO to extract coordinate and velocity trajectories from the NGSIM video files to test our models on along with the existing NGSIM trajectory data. Secondly, we identified additional parameters influencing the velocity profiles of vehicles and extracted this data from the NGSIM dataset. Finally, we then built and fine-tuned two AI models - a Transformer and LSTM model - for time-series prediction. We utilized these velocity predictions to identify anomalous points in the real-life NGSIM velocity trajectories.

2 Background Research

Time series modeling has been an extensively researched area, specifically in developing autonomous vehicles where human behaviour on traffic roads can be learnt effectively. LSTM (type of RNN) architecture was proposed first in 1997[1] and till date prove to be an extremely successful approach to time series modeling. Key area of concern with LSTM was its inability to handle long range dependencies, which was addressed largely by the famous research paper - "Attention is all you need"[2] introducing the transformer architecture, which utilised self attention to resolve this issue.

For training autonomous vehicles, preparing dataset resembling human behaviours is a major concern. The US Dept. of Transportation publicly provides **Next Generation SIMulation** (NGSIM)[3] dataset capturing multiple traffic conditions and recording trajectories of vehicles. Rich research has been conducted on this dataset, particularly "Modelling Lane-Changing Behaviour in a Connected Environment: A Game Theory Approach"[4] and "An LSTM Network for Highway Trajectory Prediction".[5] Our work largely builds upon these existing LSTM based research by using transformers and comparing performance.

3 Data description

ITS DataHub has partnered with the Federal Highway Administration's NGSIM program to make data publicly available from the NGSIM data collection efforts. The NGSIM program collected high-quality traffic datasets at four different locations, including two freeway segments (I-80 and US-101) and two arterial segments (Lankershim Boulevard and Peachtree Street), between 2005 and 2006. The vehicle trajectory data as well as the video files of the traffic are both available publicly where the trajectory data provides the precise location of each vehicle every one-tenth of a second, including detailed lane positions and locations relative to other vehicles. The table below showcases the training and testing datasets that we ran our models on. The full US-101 dataset was utilized for training.

Location	Time 1	Time 2
US-101 (training)	-	-
I-80 (testing)	4:00 - 4:15	5:00 - 05:15
PeachTree (testing)	12:45 - 13:00	04:00 - 04:15

4 Initial data analysis

4.1 Visualizing Velocity and Acceleration

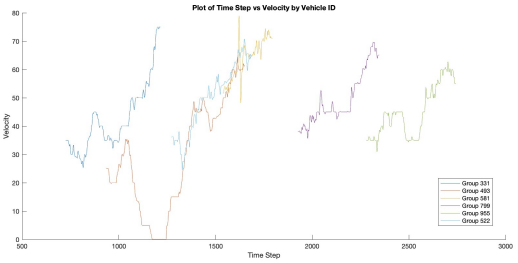
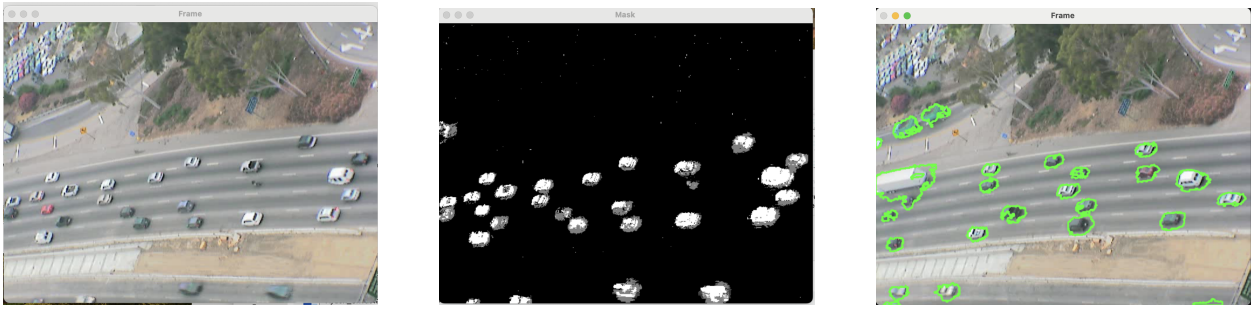


Figure 1: Time Series graph of Velocity

Our initial list of project objectives included modelling vehicle acceleration. In order to determine how feasible this goal was, we plotted the acceleration time series data along with velocity and coordinates. We observed that the velocity profile data showed more time series dependency than acceleration. We realized that the acceleration datapoints are likely to be noisy and unrealistic due to the noise in the velocity dataset. We thus choose to model velocity profiles only.

4.2 GMM for position and velocity extraction

As mentioned above, we aim to leverage CV techniques to extract the vehicle coordinate and velocity data from the NGSIM video files. We initally utilized a simple Gaussian Mixture-based Background/Foreground Segmentation Algorithm[6] to remove background objects from the videos and generate contours of moving vehicles. The exact position of the vehciles can be extracted by tracking midpoint of these contours and velocity can be measured by numerical differentiation of position profile. However, we wanted to extract the trajectories with higher precision and reduced noise, so we instead chose to use pre-trained image detection model called YOLO.



(a) Video from NGSIM (b) Background Removal (c) Contour Identification

Figure 2: Process of extracting moving object profile

5 Trajectory extraction

5.1 Modelling Coordinates and Velocity

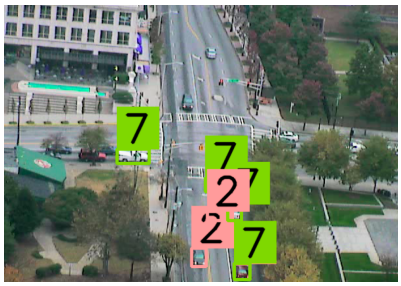


Figure 3: YOLO vehicle detection

YOLO (You Only Look Once)[7] is a popular object detection model that uses an end-to-end neural network to makes predictions of bounding boxes and class probabilities all at once. YOLO differs from the approach taken by previous object detection algorithms, which repurposed classifiers to perform detection. We used the latest version of YOLO model - version 8 from Ultralytics to identify vehicle profiles.
The selected classes of interest were car, motorcycle, bus and truck, and all videos were 10 fps. The supervision package was used to generate the frames and obtain the object detections from the YOLO results. Our code identifed the coordinates of the bounding box for multiple vehicles in each frame of the video, and found the coordinates to be the midpoint of the box. To calculate velocity, euclidean distance and frame rate was then utilized. As YOLO was unable to track every vehicle of interest in every frame, missing timestep data was filled in manually using the average difference in coordinates/velocity between tracked frames.

5.2 Modelling Other parameters

A vehicle's velocity at timestep t+1 can depend on a number of parameters in its past history up until time t. Two of these parameters include its past coordinates and velocity values. We initially extracted the coordinates + velocity of vehicles and used these as features to model velocity profiles as a proof of concept step. We then proceeded to identify which other parameters are most correlated with velocity, and build a model using those variables.

A number of other parameters can influence a vehicle's velocity as well, which we have narrowed down to the following: Direction of movement, movement of vehicle, and distance from preceding vehicle.

6 Time series prediction

Our main objective is to predict the future trajectory of a vehicle given its history; this is a time series modeling problem We look to study 2 main architectures of time-series modeling - Transformers and LSTM.

6.1 LSTM

Long Short-Term Memory Networks is a deep learning, sequential neural network that allows information to persist. It is a special type of Recurrent Neural Network which is capable of handling the vanishing gradient problem faced by RNN. LSTM is generally the go-to architecture for time-series modelling because it has less number of parameters than transformers and facilitates online learning more.

The LSTM architecture is as follows[1]

$$
\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \\
c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \\
h_t &= o_t \odot \tanh(c_t)
\end{aligned}
$$

where h_t is the hidden state at time t, ct is the cell state at time t, x_t is the input at time t, h_{t-1} is the hidden state of the layer at time $t-1$ or the initial hidden state at time o, and i_t, f_t, g_t, o_t are the input, forget, cell, and output gates, respectively. σ is the sigmoid function, and $\odot$ is the Hadamard product.

6.2 Transformers

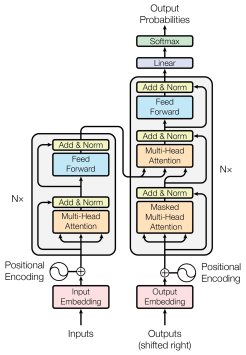


Figure 4: Transformer Architecture

Motivation: Learning Textual data and Time Series Data have similar characteristics: Past data-points have a quantifiable effect on future data-points The closer data-points to current one affect it more than those farther away.

The models[2] has an encoder-decoder structure. The encoder and decoder are composed of N=6 layers. Encoder layer has 2 sub layer as shown in figure 4 (left stack). Decoder layer (right stack) has an additional multi-head attention layer which operates on the output of input encoder later. Model also modifies the self-attention sub-layer in the decoder stack to prevent positions from attending to subsequent positions. This masking, combined with fact that the output embedding are offset by one position, ensures that the predictions for position i can depend only on the known outputs at positions less than i.

7 Results

7.1 Modelling Trajectory

We trained both Transformer and LSTM models to predict velocity profiles of each vehicle present in the full NGSIM US-101 dataset. Both models were fed with 2000+ vehicle velocity and position profiles. The input parameters are the local coordinates of the vehicle and its velocity history. We then tested both models on the NGSIM trajectory data as well as the YOLO extracted data for the Peachtree and I-80 datasets. All the test results are shown below for single vehicles, where velocity is measured in mph for the NGSIM trajectory data. It was observed that LSTM trained considerably faster than Transformer for our case. Additionally, the prediction from Tranfsormer model are slightly shifted towards right on account of sequence length parameter whereas the LSTM seem to fit the time series perfectly.

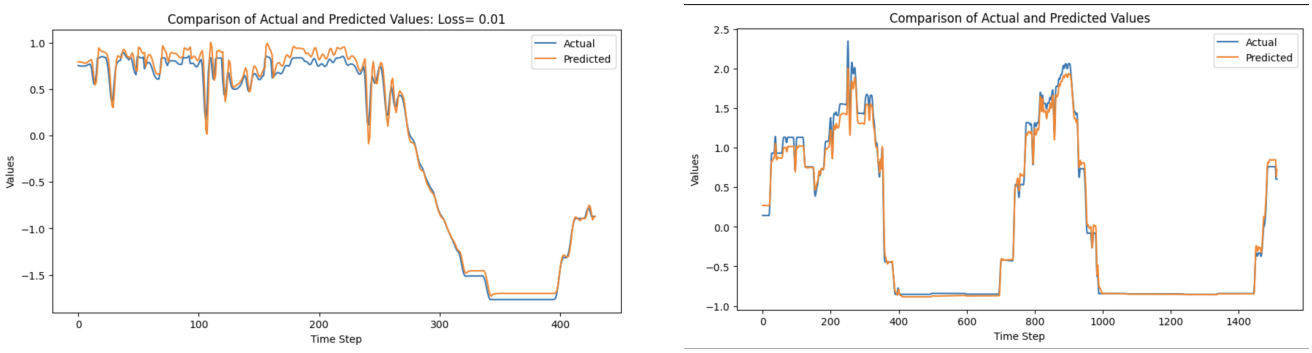


Figure 5: Transformer Prediction on Highway I-80 (Left) and Intersection Peachtree (Right)

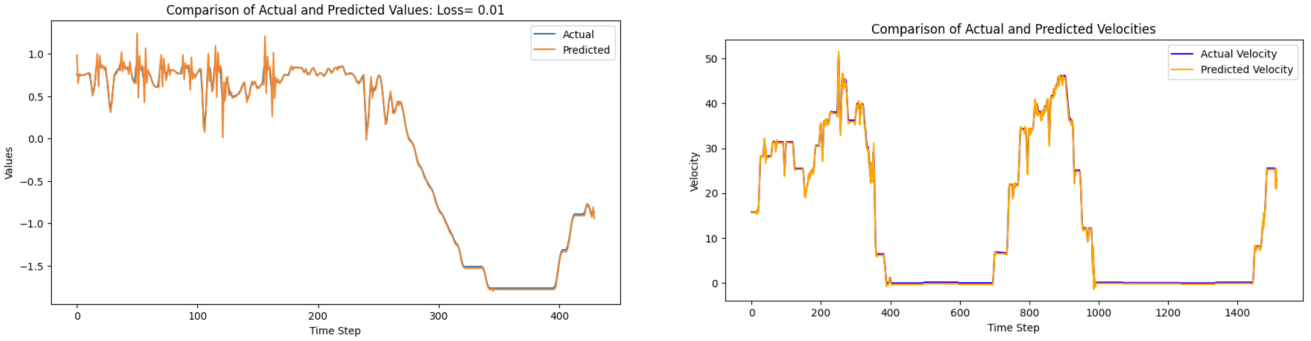


Figure 6: LSTM Prediction on Highway I-80 (Left) and Intersection Peachtree (Right)

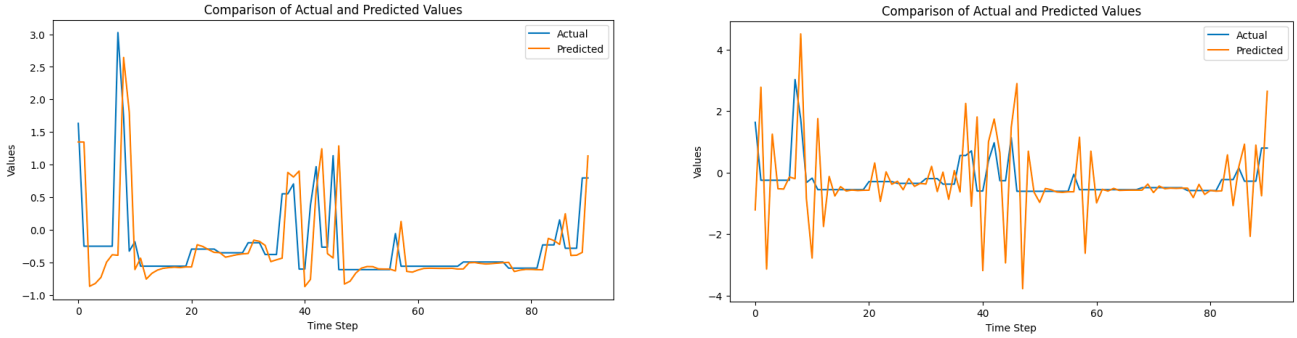


Figure 7: YOLO + Transformer (left) and LSTM (right) results on Peachtree dataset 12:45 - 1:00 (Single vehicle)

Test Loss	I-80 Street	Peachtree Street
Transformer	0.067	0.081
LSTM	0.023	0.003
YOLO+Transformer	1.065	0.313
YOLO+LSTM	1.226	1.225

Table 1: Test loss on various models and scenarios

From an loss perspective, performance of both the models is similar when tested on the I-80 and Peachtree data. We observe that the Transformer model had a higher loss across datasets (0.067 versus 0.023 and 0.081 versus 0.003) when compared to the LSTM model when tested on the NGSIM trajectory data. However, when we test the models on trajectories extracted from videos using YOLO, the Transformer model tended to perform much better than the LSTM model. (1.065 versus 1.226 and 0.313 versus 1.225). It can also be observed that the loss is greater for the YOLO results when compared to the NGSIM dataset, likely because the YOLO model failed to detect a majority of vehicles in 100 percent of the frames.

7.2 Detecting Anomalous Behavior

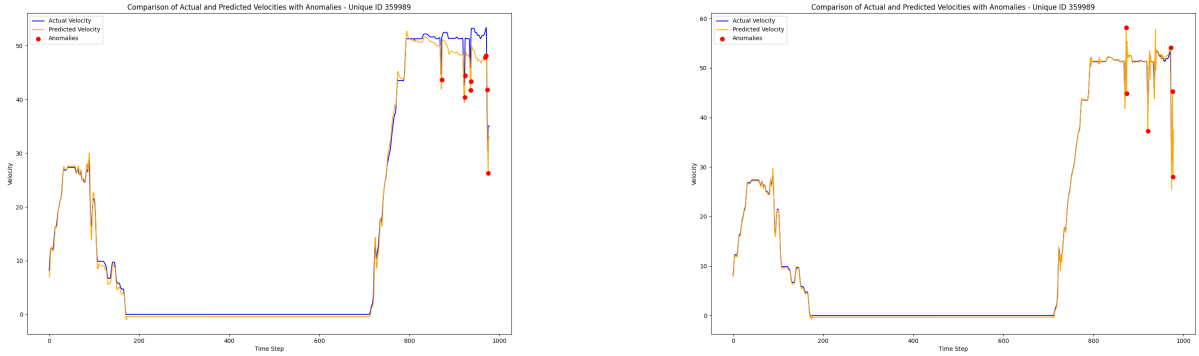


Figure 8: Anomaly Detection with Transformers (Left) and LSTM (Right) at fixed threshold - I-80

Motivated by research done on anomalous detection,[8] we utilized our predictions to identify anomalous data points in the NGSIM trajectory data. To determine our anomaly threshold, we analyse differences in actual and predicted values and identify outlier points via a boxplot. This is done on the training dataset and threshold is determined. This threshold is used to detect anomaly in test datasets.

We observe that though both the models identify outlier at the same time period, Transformers are more susceptible to identifying outliers than LSTM. Identifying outlier is important if the autonomous vehicle is to make quicker decisions regarding safety based on other vehicle behaviour.

8 Conclusion and Future work

Ultimately, our project successfully utilized the YOLO model to extract coordinate and velocity trajectories from the NGSIM video files and identified and extracted additional parameters influencing the velocity profiles of vehicles. We then built and finetuned a Transformer and LSTM model for time-series prediction. and utilized their predictions to identify anomalous points in the real-life NGSIM velocity trajectories.

Future of our work involves testing this model on an actual vehicle in a specified traffic environment with multiple vehicles. For virtual simulation, we will utilise ROS2 software libraries to simulate autonomous car in required scenario. Following steps would be taken up:

1. ROS2 environment allows 2 vehicles to talk to each other, so it is possible for ego vehicle (implementing our algorithm) to directly use velocity of opp vehicle (Human driven vehicle) and predict next steps and take decisions for itself.

2. Camera vision would be utilised to track real time opp vehicle velocity by placing camera of required specifications. This camera will be used to implement our YOLO model to extract velocity from real time vision of opp vehicle and then predict its next behaviour.

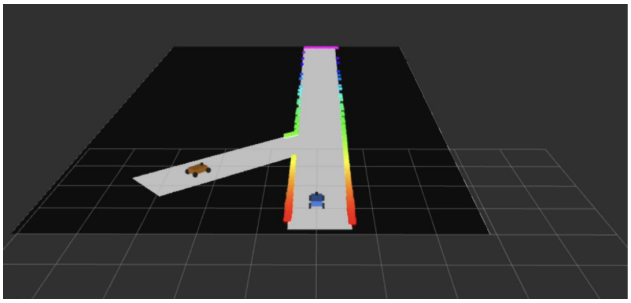


Figure 9: Sample Simulation Environment on ROS2 for Testing Multi-agent Behaviour

References

[1] Sepp Hochreiter and Jurgen Schmidhuber. Long Short-Term Memory. In *Neural Computation 9, 1735–1780 Massachusetts Institute of Technology*. 1997.

[2] Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, Polosukhin. Attention Is All You Need. In *31st Conference on Neural Information Processing Systems (NIPS 2017)*, , Long Beach, CA, USA. 2017.

[3] U.S. Department of Transportation Federal Highway Administration. Next Generation Simulation (NGSIM) Vehicle Trajectories and Supporting Data. *Provided by ITS DataHub through Data.transportation.gov*, Accessed 2024-05-01 from http://doi.org/10.21949/1504477

[4] Alireza Talebpour, Hani S. Mahmassani, Samer H. Hamdar. Modeling Lane-Changing Behavior in a Connected Environment: A Game Theory Approach In *Transportation Research Procedia*, Volume 7, 2015, Pages 420-440, ISSN 2352-1465, https://doi.org/10.1016/j.trpro.2015.06.022.

[5] Altché, Florent & de La Fortelle, Arnaud. An LSTM Network for Highway Trajectory Prediction. 2018

[6] V. Anand, D. Pushp, R. Raj and K. Das. Gaussian Mixture Model (GMM) Based Object Detection and Tracking using Dynamic Patch Estimation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*,Macau, China, 2019, pp. 4474-4481, doi: 10.1109/IROS40897.2019.8968275.

[7] V. Anand, D. Pushp, R. Raj and K. Das. You Only Look Once: Unified, Real-Time Object Detection. In *arXiv 1506.02640v5*, 2015

[8] Fan Jiang, Junsong Yuan, Sotirios A. Tsaftaris, Aggelos K. Katsaggelos Anomalous video event detection using spatiotemporal context. In *Computer Vision and Image Understanding*, 2010