Proposal for “Neural Network based Safety Metric for Autonomous Vehicles”

Haojie Zhu
Department of Civil and Environmental Engineering
University of Michigan
Ann Arbor, Michigan, USA
zhuhj@umich.edu

I. Overview

Over the past century, autonomous vehicles (AVs) have continuously captured public interest due to their potential to enhance convenience and safety in daily life. Substantial advancements in the field of AVs have been driven by developments in computer science, particularly with the advent of convolutional neural networks (CNNs). These networks enable vehicle cameras to efficiently recognize and classify obstacles, as highlighted in [1].

Despite these advancements, public confidence in AVs remains tepid, primarily due to safety concerns. For instance, a notable incident in 2023 involved a Robotaxi from Cruise that struck and dragged a pedestrian, underscoring the gap between current AV performance and public expectations. Enhancing the safety performance of AVs is paramount, and safety metrics play a crucial role in this endeavor. An accurate safety metric module can promptly alert autonomous systems to take emergency actions, potentially averting accidents. Moreover, such a module can identify critical driving scenarios essential for the further development of AV technologies.

Current research in AV safety metrics predominantly involves predicting the behaviors of background vehicles (BVs) and AVs based on predefined rules to assess danger levels. This paper will delve into these existing methodologies in greater detail. Rule-based safety metrics often lean towards over-conservatism by considering all possible behaviors of background vehicles (BVs), which can adversely affect overall traffic mobility. Additionally, the inherent limitations of rule-based systems may cause them to overlook rare yet safety-critical scenarios.

To address these challenges and improve the accuracy of AV safety metrics, this project explores the implementation of deep learning algorithms, specifically a neural network-based safety metric. Given the suitability of Multilayer Perceptrons (MLPs) for common classification and regression tasks [2], we propose an MLP-based safety metric to evaluate safety-critical situations under current conditions. This work focuses on multi-lane highway scenarios, utilizing a collision-inclusive simulation with Simulation of Urban MObility (SUMO) [3] to construct the dataset. The developed safety metric demonstrates superior accuracy and precision compared to existing rule-based metrics, including the Safety Metric based on the Assessment of Risk (SMAR) [4], Model Predictive Instantaneous Safety Metric (MPrISM) [5], Pegasus Criticality Measure (PCM) [6], and time-to-collision (TTC) [7].

II. Prior Work

Many researchers are focused on developing accurate safety metrics for autonomous vehicles. SMAR leveraged naturalistic driving data to create a realistic probabilistic behavior model for human drivers. This model can predict the collision probability for any given traffic state by considering real-world driving characteristics. However, building such a realistic behavior model is challenging, especially when trying to model driving behavior in safety-critical scenarios due to data limitations.

Another approach, MPrISM, determines the safety status of the ego vehicle by considering the worst-case safety scenario for a given traffic snapshot. This method assumes that background vehicles will take the most aggressive actions to cause a crash, and it calculates how the ego vehicle can avoid such an accident. The danger level is then determined based on the ratio between the ego vehicle’s behavior and its kinematic limits. The PCM algorithm uses Model Predictive Control (MPC) to control the ego vehicle and calculate the optimal acceleration decision while assuming that background vehicles maintain their speed and heading. A scenario is identified as safety-critical when the required acceleration exceeds a certain threshold. The TTC method, on the other hand, assumes that the ego vehicle maintains its current state, such as velocity and heading, and estimates the minimum time until a collision with other vehicles occurs. The effectiveness of these methods heavily depends on their underlying assumptions. For instance, MPrISM could be overly conservative, as other vehicles are unlikely to always take the most aggressive actions in normal situations. Both PCM and TTC assume that background vehicles will keep their current states, which is not always the case in real-world scenarios.

Given the limitations of existing safety metrics, the recent availability of extensive naturalistic driving data, including crashes and other rare events, has generated interest in applying machine learning methods to develop more accurate and general safety metrics for autonomous vehicles. This project aims to investigate the advancements of neural network-based safety metrics compared to existing methods.

III. Preliminary Results

To simplify the problem, I focus on a scenario involving a three-lane highway, as illustrated in Fig. 1. To replicate a naturalistic driving environment (NDE), the behavior of BVs is modeled using real-world naturalistic driving data (NDD). More details on the NDE modeling method used in this study are outlined in [9]. To efficiently generate safety-critical trajectories, I utilized the naturalistic and adversarial driving environment (NADE) [10], which produces trajectories that include collisions. The ego vehicle is controlled using three different autonomous vehicle (AV) algorithms: one based on the Intelligent Driving Model (IDM) [11] and the Minimizing Overall Braking Induced by Lane Change (MOBIL) model [12], and the other two based on reinforcement learning methods [13].

Fig. 1. The studied three-lane highway with a length of 1200 meters. The green symbol represents a background vehicle, while the red symbol denotes the ego vehicle. The ego vehicle initiates at one-third of the highway and drives for an additional 400 meters. The length and width of all vehicles are 5m and 1.8m, respectively. The lane width is 4m.

A total of 1,291 hours of simulation data were collected with a time resolution of 0.1 seconds, during which 3,797 crashes were recorded. The dataset was then divided into three parts: 20% for validation, 20% for testing, and the remaining 60% for training. Using the logged trajectory data, the evaluation framework proposed in [14] was employed to determine whether a collision involving the ego vehicle was unavoidable at the studied moments. Essentially, an optimization problem is formulated to compute if there exists an evasive trajectory for the ego vehicle, given all near-future logged trajectory information of the BVs. If no feasible evasive trajectory existed, the situation was labeled as collision unavoidable (label of 1), whereas the remaining steps were labeled as 0. Data points with a positive label only accounted for 0.0144% of all the data, presenting a challenge for training a neural network-based classifier.

To begin, I selected MLPs as the foundational model. As depicted in Fig. 1, the proposed neural network comprises a frequency encoding layer, an MLP with a Batch Normalization backbone, a prediction layer, and a sigmoid function layer. The raw input includes vehicle states such as position, heading, and velocities. Before feeding the state vectors into the MLP with a Batch Normalization backbone, we employ the frequency encoding method outlined in [15], which uses a set of sine and cosine basis functions to project the vectors into a high-dimensional space, thereby improving the capture of high-frequency variations in the state spaces. Batch Normalization is integrated into the MLP neural network to stabilize the training process and enhance performance [16]. Finally, a linear prediction layer and a sigmoid function layer are used to output the safety criticality of the current scenario.

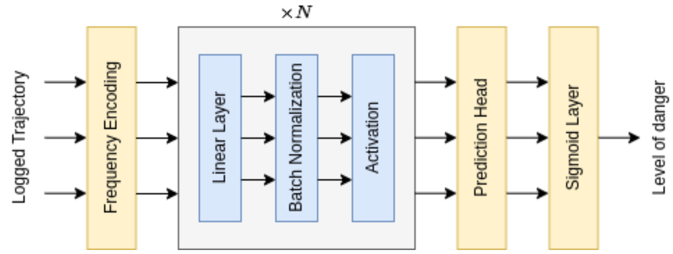

Fig. 2. Overall structure of the neural network.

In this project, I will employ various tools learned from STATS 507, such as NumPy, Torch, Pandas, SciPy, and Matplotlib, to analyze the dataset, train the neural network, and visualize the results. Furthermore, I will explore additional tools like Gurobi, GAMS, and MATLAB for adapting the existing safety metrics to the studied scenario.

IV. Project Deliverables

This project aims to develop a cutting-edge neural network-based safety metric that surpasses existing safety metrics such as SMAR, MPrISM, PCM, and TTC in determining the level of danger of any given scenario. The following deliverables will be produced as part of this project:

1) Source Data: An annotated dataset used for training, testing, and validation of the neural network model will be provided. Data will be preprocessed and organized in a manner that ensures ease of use and comprehensiveness for future research and development.
2) Neural Network Based Safety Metric: A novel safety metric leveraging neural network methodologies will be delivered. It is designed for superior accuracy, reliability, and applicability in various safety-critical environments.
3) Performance Evaluation Report: Comprehensive documentation and analysis comparing the performance of the new safety metric against existing safety metrics (i.e., SMAR, MPrISM, PCM, and TTC) will be provided. Detailed performance including precision, recall, and computational efficiency will be recorded.
4) Source Code: Fully documented source code for the neural network model and any supporting algorithms and scripts will be delivered. The code will be modular and optimized for extensibility to facilitate future enhancements and adaptations.

V. Timeline

I will follow the following schedule to finish this project:

- Week 1
    1) Collect the trajectory data from a collision-inclusive multi-lane highway simulation using SUMO
    2) Understand the trajectory data and construct the training and validation dataset
    3) Start working on the Proposal of the project
- Week 2

1) Adapt SMAR, MPrISM, PCM, and TTC algorithms to the studied scenarios
2) Evaluate the performance of SMAR, MPrISM, PCM, and TTC using the collected trajectory data

- Week 3
  1) Complete the Proposal
  2) Design the structure of the neural network based safety metric and conduct some preliminary experiments on the training dataset
- Week 4
  1) Train the neural network based safety metric and compare its performance with other safety metrics (i.e., SMAR, MPrISM, PCM, and TTC)
  2) Prepare the source code
- Week 5
  1) Summarize the experiment results
  2) Complete the final report

REFERENCES

[1] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 1907-1915, 2017.

[2] F. Murtagh, "Multilayer perceptrons for classification and regression," Neurocomputing, no. 5-6, pp. 183-197, 1991.

[3] D. Krajzewicz, "Traffic simulation with SUMO–simulation of urban mobility," Fundamentals of traffic simulation, pp. 269-293, 2010.

[4] S. Feng, H. Sun, X. Yan, H. Zhu, Z. Zou, S. Shen, and H. X. Liu, "Dense reinforcement learning for safety validation of autonomous vehicles," Nature, pp.620-627, 2023.

[5] B. Weng, S.J. Rao, E. Deosthale, S. Schnelle, and F. Barickman, "Model predictive instantaneous safety metric for evaluation of automated driving systems," IEEE Intelligent Vehicles Symposium (IV), pp. 1899-1906, 2020.

[6] P. Junietz, F. Bonakdar, B. Klamann, and H. Winner, "Criticality metric for the safety validation of automated driving using model predictive trajectory optimization," International Conference on Intelligent Transportation Systems (ITSC), pp. 60-65, 2018.

[7] D. N. Lee, "A theory of visual control of braking based on information about time-to-collision," Perception, 5(4):437–459, 1976.

[8] E. Camacho and C. Alba, "Model predictive control," Springer Science and Business Media, 2013.

[9] X. Yan, S. Feng, H. Sun and H. X. Liu, "Distributionally consistent simulation of naturalistic driving environment for autonomous vehicle testing," arXiv:2101.02828, 2021.

[10] S. Feng, X. Yan, H. Sun, Y. Feng, and H. X. Liu, "Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment," Nature communications, vol. 12, no. 1, pp. 748, 2021.

[11] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," Physical review E, 62(2), 1805, 2000.

[12] A. Kesting, M. Treiber, and D. Helbing, (2007). "General lane-changing model MOBIL for car-following models," Transportation Research Record, 1999(1), 86-94, 2007.

[13] S. Feng, H. Sun, X. Yan, H. Zhu, Z. Zou, S. Shen, and H. X. Liu, "Dense reinforcement learning for safety validation of autonomous vehicles," Nature, 615(7953), 620-627, 2023.

[14] X. Yan, S. Feng, D. J. LeBlanc, C. Flannagan and H. X. Liu, "Evaluation of Automated Driving System Safety Metrics With Logged Vehicle Trajectory Data," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 8, pp. 8913-8925, Aug. 2024.

[15] B. Mildenhall, et al., "NeRF: representing scenes as neural radiance fields for view synthesis," In: European conference on computer vision, pp.405–421, 2020.

[16] S. Ioffe, and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in Proceedings of the 32nd International Conference on Machine Learning, 2015.