Predicting Safety-Critical Misbehaviours in Autonomous Driving Systems using Autoencoders

Andrea Stocco, Michael Weiss, Marco Calzana, Paolo Tonella
Università della Svizzera italiana, Lugano, Switzerland
andrea.stocco|michael.weiss|marco.calzana|paolo.tonella@usi.ch

ABSTRACT

This extended abstract paper summarizes our ongoing research related to improving the dependability of DNN-based autonomous driving systems. Particularly, we address the problem of recognizing unexpected execution contexts with the purpose of predicting potential safety-critical misbehaviours. Our approach SelfOracle is based on a novel concept of self-assessment oracle, which monitors the DNN confidence at runtime, to predict unsupported driving scenarios in advance. SelfOracle uses autoencoder- and time series-based anomaly detection to reconstruct the driving scenarios seen by the car, and to determine the confidence boundary between normal and unsupported conditions. Our evaluation using three self-driving car models shows promising results against a diverse set of simulated anomalous driving contexts.

1 RESEARCH PROBLEM AND MOTIVATION

Autonomous driving systems work by training Deep Neural Networks (DNNs) on a multitude of sensor data collected during in-field driving sessions. To date, the best performing cars by Waymo/-Google, Tesla, and Uber have shown great results in driving several hundreds of miles without any human intervention [1, 2]. However, training sets are by construction limited to the observed situations. Hence, they hardly contain all possible driving scenarios that can be met everyday and that can be covered in the testing phase.

The problem is relevant because unseen execution contexts are unknown at training time, and an autopilot which is not trained in such situations is likely to fail, if they deviate remarkably from the training data, in the worst case leading to catastrophic failures (e.g., fatal crashes). We aim to build an additional self-driving component designed to monitor the novelty of the context where an autopilot is executing. Indeed, an accurate misbehaviour predictor is a necessary prerequisite to apply preventive or reactive techniques (e.g., transferring the control to the human driver, in case of driving assistance systems).

The goal of this research is to predict misbehaviours in autonomous driving systems. We make use of driving simulation platforms to observe and test the system in operation during the occurrence of unseen simulated scenarios. A predictor, previously trained to recognize nominal conditions, must be able to detect the change in the driving scenarios timely enough to enable preventive countermeasures to take place.

Our approach is based on the reconstruction error of autoencoders as a black-box confidence metric. Our focus is on simulation-based scenarios, for which it is possible to collect rich and precise information on the vehicle's misbehaviors safely. Moreover, we designed simulation-based test scenarios that allow testing SelfOracle on a large number of challenging conditions (e.g., adverse weather, or low light conditions).

2 EXPERIMENTAL APPROACH

Predicting mis-behaviours during the motion of a vehicle makes dataset analysis in autonomous driving research more challenging than just identifying single underrepresented images in the training set. To face this challenge, our approach combines a reconstruction-based anomaly detector with a predictive model of normality based on time series analysis.

Our experimental approach consists of several steps. ❶ We record data representative of the behaviour of the car when driving in nominal conditions. ❷ We fit a probability distribution of the nominal behaviours and set a confidence level (i.e., a threshold) that defines the acceptable false positive rate. ❸ We inject anomalous/unseen conditions in the simulator. ❹ We re-execute the self-driving car, recording each failure of the self-driving component, namely out-of-bounds and collisions. ❺ We evaluate whether our predictive model is able to signal the occurrence of such failures in advance. We investigated the effectiveness of our approach in the Udacity simulator [5], a popular cross-platform environment for self-driving agents developed with Unity.

❶ Normal Behaviour Reconstructor. The first step consists in retrieving a model of normality from the training driving scenarios. The training set captures the visual input stream of the self-driving car under nominal situations. We considered three self-driving car models that are run over a set of three different tracks. They were trained with the goal of obtaining reliable models that experience no misbehaviours when executed in nominal situations (i.e., sunny weather conditions). Overall, our training set contains 124,638 training images.

Then, we trained our mis-behaviour predictor on such "normal" instances. In particular, we used as reconstructors four autoencoders: (1) *SAE* (simple autoencoder with a single hidden layer), (2) *DAE* (deep five layers fully-connected autoencoder), (3) *CAE* (convolutional autoencoder alternating convolutional and max-pooling layers), and (4) *VAE (variational autoencoder)*. We chose as a baseline DeepRoad$_{IV}$, an input validator based on feature extraction and PCA [6].

❷ Probability Distribution Fitting. After building a model of normality for the reconstruction errors collected in nominal driving conditions, we determined a threshold θ that brings the expected false alarm rate in nominal conditions below a user defined threshold ϵ (e.g., 0.05 or 0.01). In detail, we fit a Gamma distribution of the mean squared errors (MSE) produced by the autoencoders [3]. We use probability distribution fitting to obtain a statistical model of normality, rather than using the raw reconstruction error frequency distribution, because high error values are rare and may have zero frequency, while the tail of a Gamma distribution is zero only asymptotically. In other words, the estimated false alarm rate would be incorrectly assumed to be equal to zero when only a few,

or even no data points, are observed on the right of the chosen threshold.

❸ **Unseen Conditions Generation.** We implemented two additional components within the Udacity simulator, namely, an *unexpected context generator*, and a *collision/OBE detection system*. The former gradually injects unseen conditions in autonomous driving mode (i.e., conditions diverse from the training mode's defaults). Instances of these situations deal with *illumination* (day/night cycle) or *weather* (rain, snow, fog), as well as their possible combinations. The latter records any unwanted interaction of the self-driving car with the environment during testing (e.g., collisions, or car driving off track). The result is a set of labeled images that we can use to experiment the effectiveness of SelfOracle at anticipating such unexpected scenarios.

❹ **Misbehaviour Prediction.** Our evaluation data consist of 72 simulations. We ran all tested self-driving car models on all available tracks under all conditions. Overall, we obtained a dataset of 778,592 images. We split the evaluation set into *windows of consecutive frames*, which we labelled as either anomalous or normal. The goal of SelfOracle is maximizing the prediction of shortly-following misbehaviors in anomalous windows (true positives), while minimizing the false alarms, i.e., wrong misbehavior predictions in normal windows (false positives). In our experiments, we set the length of normal/anomalous windows to 30 frames, which is ≈3s in Udacity. As metrics for evaluation, we used the standard classification metrics: TPR, FPR, F1-score. Moreover, we used two threshold-independent metrics, namely AUC-ROC (area under the curve of the Receiver Operating Characteristics), and AUC-PRC (area under the Precision-Recall curve).

2.1 Implementation

We implemented our approach in a publicly available Python tool called SelfOracle [4]. The tool supports Self-Driving Car (SDC) models written in Keras 2.2.4, and has been experimented on the Udacity simulator for self-driving cars [5]. SelfOracle can be used for monitoring the condition of a self-driving car, in order to identify context changes which may be indicative of a future failure.

3 RESULTS AND CONTRIBUTIONS

A comprehensive empirical validation of our approach is described in our full paper [3]. Here, we briefly report the main findings.

3.1 Evaluation Summary

We evaluate our framework on three existing DNN-based SDCs: Nvidia's DAVE-2, Epoch, and Chauffeur [3]. To collect the evaluation data, we executed 72 simulations (2 laps each) in autonomous mode (3 SDC x 8 conditions x 3 tracks). Simulated conditions were: day/night cycle, rain, snow, fog, day/night cycle + rain, day/night cycle + snow, day/night cycle + fog. We use the input validation technique of DeepRoad$_{IV}$ [6] as baseline for SelfOracle.

Effectiveness. In our experiments, the best reconstructors are VAE and SAE, with comparable overall performance. At $\epsilon = 0.05$, VAE predicts correctly 589/765 misbehaviours (77%), with 84/765 false alarms (11%) due to adverse conditions that were not that extreme to make the system fail. This was expected, since we are measuring FPR in tracks with injected anomalies. DeepRoad$_{IV}$ predicts correctly 252/765 misbehaviours (33%), with 76/765 false alarms (10%). Thus, VAE detected 337 more misbehaviours, with a comparable false alarm rate. At $\epsilon = 0.01$, SAE predicts correctly 451/765 misbehaviours (59%), with 38/765 false alarms (5%). DeepRoad$_{IV}$ predicts correctly 153/765 misbehaviours (20%), with 46/765 false alarms (6%). Thus, SAE detected 298 more misbehaviours, again with a comparable false alarm rate.

Prediction. In our experiments, all configurations of SelfOracle are able to predict, on average, an upcoming misbehaviour up to 60 frames (around 6 s) in advance.

Comparison. To summarize, in our experiments SelfOracle has shown to be more effective than DeepRoad$_{IV}$ at predicting misbehaviours. Results of AUC-PRC and AUC-ROC show significant improvements across all thresholds, regardless of the technique being used and the reaction period considered. Concerning the performance, in our experiments, the autoencoders took ≈3 ms per prediction whereas DeepRoad took ≈45 ms per prediction (+1400% increment). While both runtime measures may seem acceptable in practical scenarios, it is worth remembering that DeepRoad$_{IV}$ requires to dramatically sub-sample the training set available for the experiments to achieve such execution times. Indeed, only few hundreds images can be used, because the technique behind DeepRoad$_{IV}$ is computationally very expensive. Hence, differently from our approach, it is also quite unlikely to scale to training datasets used by industry manufacturers.

3.2 Contributions

Our major contribution is the design of a predictive model for failure estimation in autonomous driving systems based on a black-box confidence measure and probability distribution fitting. Our approach uses only input information to predict failures, which makes it independent from the used DNN architecture and applicable in principle to any self-driving car model. We implemented our approach in the publicly available tool SelfOracle [4].

Predicting failures is a prerequisite for enabling (semi-)automated healing techniques. In our preliminary experiments we evaluated safety requirements violations such as collisions, even though other driving requirements (e.g., smoothness of driving) can be tested thanks to our framework. As a follow-up, we plan to apply online confidence monitoring based on white-box metrics, with the potential for hybridization.

REFERENCES

[1] BGR Media, LLC. 2018. Waymo's self-driving cars hit 10 million miles. https://techcrunch.com/2018/10/10/waymos-self-driving-cars-hit-10-million-miles. Online; accessed 18 August 2019.

[2] Vinton G. Cerf. 2018. A Comprehensive Self-driving Car Test. *Commun. ACM* 61, 2 (Jan. 2018), 7–7. https://doi.org/10.1145/3177753

[3] Andrea Stocco, Michael Weiss, Marco Calzana, and Paolo Tonella. 2020. Misbehaviour Prediction for Autonomous Driving Systems. In *Proceedings of 42nd International Conference on Software Engineering (ICSE '20)*. ACM.

[4] SelfOracle 2020. Misbehaviour Prediction for Autonomous Driving Systems. https://gitlab.dev.si.usi.ch/USI-INF-Software/precrime/tree/master/selforacle.

[5] Udacity. 2017. A self-driving car simulator built with Unity. https://github.com/udacity/self-driving-car-sim. Online; accessed 18 August 2019.

[6] Mengshi Zhang, Yuqun Zhang, Lingming Zhang, Cong Liu, and Sarfraz Khurshid. 2018. DeepRoad: GAN-based Metamorphic Testing and Input Validation Framework for Autonomous Driving Systems. In *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering (ASE 2018)*. ACM, New York, NY, USA, 132–142. https://doi.org/10.1145/3238147.3238187