Robust Detection of Anomalous Driving Behavior

Matthias Matousek*, Mahmoud Yassin†, Ala'a Al-Momani*,
Rens van der Heijden*, Frank Kargl*

*Institute of Distributed Systems, Ulm University
matthias.matousek@uni-ulm.de, alaa.al-momani@uni-ulm.de
rens.vanderheijden@uni-ulm.de, frank.kargl@uni-ulm.de

†German University in Cairo
mahmoud.ahmed-yassin@student.guc.edu.eg

***Abstract*—Driving behavior is a major factor in traffic safety applications. Abnormal behavior, such as extremely aggressive or passive driving, can endanger both the driver and other traffic participants. Most driving behavior analysis approaches to date rely on classification, which requires labeled data for both normal driving and various types of anomalous behavior. We propose an approach that detects anomalous driving patterns based on outlier detection, which does not require such data. Apart from the required data set, existing approaches have difficulties dealing with changing behavior that overlaps with normal behavior (e.g., aggressive drivers still stop in traffic jams). We introduce a post-processing step that significantly improves results in this regard. The approach is evaluated using simulations based on the realistic LuST traffic scenario, and shows reliable detection of anomalous vehicles with low false positive rates.** [1]

I. Introduction

Safety of humans is a critical goal, especially in the automotive context. Many techniques have been developed in order to enhance and maintain individuals' safety. One of the most recent developments ones is vehicular communication, that allows vehicles to exchange information both among each other in form of vehicle-to-vehicle (V2V), as well as between the vehicles and a backend infrastructure in form of vehicle-to-infrastructure (V2I) communication. In addition to enhancing safety, vehicular communication has been developed aiming for better traffic management and efficiency. Essential for these applications are Cooperative Awareness Messages (CAMs), which are broadcast messages in which each vehicle communicates its current status, including speed, acceleration, heading, and location information. CAMs enable numerous safety applications, such as collision avoidance and hidden intersection warnings.

In this paper, we introduce a mechanism that utilizes the data contained in CAMs to detect anomalous driving behavior that has the potential to risk other road users, including, e.g., other drivers and passengers, pedestrians, and cyclists. The anomalous behavior we aim to detect ranges from an unusually slow vehicle on a highway to an aggressively driving vehicle that, e.g., frequently changes its lane, often exhibits sudden speed and acceleration changes, or maintains unusually small distances to other vehicles. We argue that any of these anomalies can pose a significant safety risk to road users, and thus our goal is to detect any deviation from normal behavior, rather than classifying the behavior itself. Therefore, we focus on the anomaly detection phase in this paper, and leave the post-detection phase — i.e., notifying the road users — to be addressed in future research.

Specifically, our contribution consists of three parts. First, we provide a comparison of different machine learning techniques for the application of anomalous driving behavior detection. Second, to provide a suitable analysis, we also evaluate the relevance of individual features, i.e., what data is taken as input for the machine learning algorithms. Finally, we introduce a methodology on how to generate a suitable and sufficiently large simulated data set, based on the widely-used LuST scenario [1] that is considered to be realistic normal behavior.

The machine learning techniques we discuss can be applied to issue warnings of anomalous behavior, and can do so either in a completely distributed fashion (V2V), or in a partially centralized fashion (V2I/I2V). The warnings themselves have numerous applications, and can be used not only for regular drivers, but also for autonomous driving applications. Specifically, the fact that some vehicles drive aggressively can be used to fine-tune prediction algorithms, further increasing their reliability.

Our approach is a machine learning-based anomaly detection mechanism that utilizes CAMs in vehicular networks in order to detect anomalous and aggressive-like driving behavior to enhance safety. In our analysis, we included three state-of-the-art unsupervised and semi-supervised machine learning algorithms:

1) k-Nearest Neighbors (k-NN) [2]
2) One-class Support Vector Machine (SVM) [3], [4]
3) Isolation Forest (iForest) [5], [6]

Unlike other approaches, using unsupervised and semi-supervised anomaly detection is not tailored to find only specific deviations of features (e.g., exceeding the speed limit), but can distinguish between overall normal and anomalous behavior, where the latter deviates significantly and thus poses a threat to traffic safety. Our mechanism is privacy-friendly as it works with anonymized data where fingerprinting the driver is not needed.

In Section II, we differentiate our work from previous approaches, such as driver fingerprinting, and further present the detection mechanisms upon which our system is based. In Section III we discuss the various phases of our scheme in detail, including data acquisition, preprocessing, outlier detection, and post-processing. We evaluate our architecture and discuss the implications in Section IV. Finally, we conclude our work and address future work possibilities in Section V.

II. Related Work

A. Driver and Vehicle Behavior

Identifying drivers by their vehicular behavior is referred to as *driver fingerprinting* and is a prime area of research focus, such as the work done by Enev et al. [7], showing that drivers can be recognized reliably by their driving behavior with only the sensor data which is readily available via a vehicle's on-board-diagnostics (OBD) port. Similarly, Van Ly et al. [8] used a vehicle's inertial sensors from the CAN bus to build a profile of the driver to provide proper feedback to reduce the number of dangerous car maneuvers. Whereas Saiprasert et al. [9] proposed an algorithm that focuses on the prospect of using minimal driving behavior signals in a V2V environment in order to remotely identify drivers. The algorithm presented shows that high identification accuracy can be achieved using limited signals.

The previously mentioned work is concerned with collecting data from the drivers' environment, then using some sort of learned model to classify and identify a specific driver within a set of known drivers in a supervised manner. Another area of research aims to classify driving styles, rather than individual drivers. For example, a rapid pattern recognition approach to characterize a driver's behavior, specifically moderate and aggressive driving behavior, is proposed in [10]. Their clustering-based SVM (kMC-SVM) method is capable of reducing recognition time and improving the recognition of driving styles.

B. Anomaly Detection

In our work, we expand the scope to consider different anomaly detection techniques and how each could recognize a vehicle's behavior and detect if it is behaving normally or exhibiting abnormal behavior using minimal privacy-preserving features. Here we briefly describe the techniques included in our analysis.

Nearest Neighbor (NN) analysis is one of the most common anomaly detection techniques, and has been invaluable when it comes to outlier detection. In a k-NN approach, data points are analyzed with respect to their neighborhood and an outlier score is assigned accordingly. The main assumption is, that normal data points exist alongside several closely related neighbors, whereas outliers tend to situate further away from other data points. In our evaluation, we are using the distance-based global k-NN, as implemented by Amer and Goldstein [2].

A supervised methodology for the one-class SVM, was proposed by Xu et al. [3]. A suppression mechanism is achieved by introducing a new variable η which represents an estimate that a certain point is normal. This variable controls the portion of slack variables that is going to contribute to the minimization objective. Thus an outlying point would have an η set to zero. Ideally, outliers would have small values of η and would thus have little or no contribution to the decision boundary. Results showed that the η enhancement provided by Amer et al. [4] for the unsupervised one-class outperformed the traditional one-class SVM and enhanced robust one-class SVM. We chose to work with the unsupervised Eta one-class SVM, due to its novel and promising performance.

Liu et al. [5], [6] proposed a new method for anomaly detection called iForest. In their work they suggest and prove that anomalies are susceptible to a mechanism called *isolation*, which takes advantage of two main characteristics of anomalies: (1) Anomalies are a minority amongst the available data space; and (2) anomalies exhibit attribute-values that are much different from normal instances.

Lui et al. implemented this mechanism using a binary tree structure called Isolation Tree (iTree), which can effectively isolate instances. As a result of the susceptibility of anomalies towards isolation, outliers tend to be isolated closer towards the root of the iTree. Whereas normal data instances are likely to be isolated towards the end of it. An iForest is essentially an ensemble of iTrees for a given dataset. The iTrees are randomly generated by adding random splits to the decision tree, until the instances are successfully isolated, resulting in a forest of trees where outlier instances have a shorter average path length.

In contrast to k-NN or SVMs, an iForest is not a distance-based approach, but rather analyzes each feature independently. This overcomes some of the disadvantages of distance-based detection mechanisms, such as their performance decrease for high input dimensions. We chose to work with iForests due to their robust performance in regard to their outlier detection capabilities as well as resource requirements.

III. Architecture

A. Dataset

Due to the difficulty in obtaining an adequately large data set with labeled driving behavior, as well as in order to perform repeatable controlled experiments, we have simulated both normal and anomalous driving behavior. In the future, we plan to extend our test to real-world data.

The widely used Simulation of Urban MObility (SUMO) toolkit enables microscopic traffic simulation, i.e., simulating traffic at a per-vehicle level. The Luxembourg SUMO Traffic (LuST) scenario [1] provides a realistic setting and enables other authors to reproduce our results. This scenario is based on the city of Luxembourg and contains almost 300,000 vehicles in a simulated road network with about 2,300 nodes and 5,900 edges with a total length of 931.12 km. There are several vehicle classes (such as passenger cars and city buses) which are being simulated over a period of 24 hours.

TABLE I: Parameters of normal and abnormal vehicle types.

vType	*accel*	*decel*	*minGap*	*sigma*	*maxSpeed*	*speedFactor*	*speedDev*	*impatience*	*tau*	*sub-lane Model*
normal0	2.6	4.5	1.5	0.5	70	–	0.1	–	–	–
normal1	3	4.5	1.5	0.5	50	–	0.1	–	–	–
normal2	2.8	4.5	1.0	0.5	50	–	0.1	–	–	–
normal3	2.7	4.5	1.5	0.5	70	–	0.1	–	–	–
normal4	2.4	4.5	1.5	0.5	30	–	0.1	–	–	–
normal5	2.3	4.5	2.5	0.5	30	–	0.1	–	–	–
aggressive0	7	8	0.5	0.1	140	1.2	0.1	–	0.05	–
aggressive1	8	9	0.5	0.2	110	1.2	0.1	1	0.1	lcPushy
aggressive2	8	9	0.3	0.1	100	1.3	0.1	1	0.01	lcAsserative
aggressive3	9	8	0.2	0.2	105	1.2	0.1	1	0.1	lcAsserative w/ lcPushy
aggressive4	10	9.5	0.7	0.1	130	1.2	0.2	1	–	lcAsserative w/ lcPushy
aggressive5	9.5	10	0.8	0.1	140	1.2	0.1	1	–	–
aggressive6	8	10	0.5	0.1	120	1.3	0.1	–	–	–

In our setup, We consider the different vehicle classes from the LuST scenario as normal driving behavior, and insert additional classes with modified parameters to represent aggressive drivers. The particular modified parameters are acceleration, deceleration, minimum gap, maximum speed, speed factor, speed deviation, impatience, and the sub-lane model [2].

The modified vehicles are inserted during peak traffic hours, in order to maximize interaction with normal vehicles. We define 7 new vehicle classes with parameters as shown in Table I. Our modifications are meant to increase the vehicle's aggressiveness. Between 8:00 AM and 8:05 AM, 14 of such vehicles are injected in the simulation, while approximately 70 normal vehicles are considered in comparison.

We collect the data from SUMO's several output files and derive features as explained in the following section.

B. Preprocessing

SUMO provides various output information about individual vehicles as well as the driving environment, such as vehicle speed, coordinates, acceleration, emissions, speed limits, slope of the road, occupancy of the current lane, and average speed of the vehicles on the current lane.

To utilize this data with our selected algorithms, we preprocess it in normalization and features extraction steps. Normalization is required, as some anomaly detection algorithms rely on distance metrics [11].

To capture behavior changes over time, we employ a sliding window approach: the window size is 3 time steps of the SUMO output (3 seconds in our case), within which all features are aggregated. The window is then moved to the next time step, resulting in overlapping windows. From each window, we collect the following features:

- the average speed over the duration of the window,
- speeding, i.e., the ratio of speed to the speed limit of the current lane of the vehicle,
- the acceleration of the vehicle aggregated and averaged over the duration of the window,
- the number of lane changes that the vehicle performs, and
- the minimum distance to the leading vehicle (minimum gap).

C. Outlier Detection

We applied and tested three outlier detection algorithms on the dataset. These were the η SVM, the k-NN algorithm and the iForest, as described in Section II.

The SVM and k-NN are distance-based algorithms, i.e., they work by analyzing the distances between samples in the feature space. The iForest, on the other hand, goes over every dimension of the input features separately and generate random decision trees, that isolate individual samples.

All three methods can give an anomaly score for input samples after they have been fitted with a dataset. A larger score corresponds to a more anomalous instance. We use both the anomaly score and the actual detection of abnormal driving behavior in our post-processing phase. Note that the scores of SVM, k-NN and iForest are not in the same range and are not directly comparable.

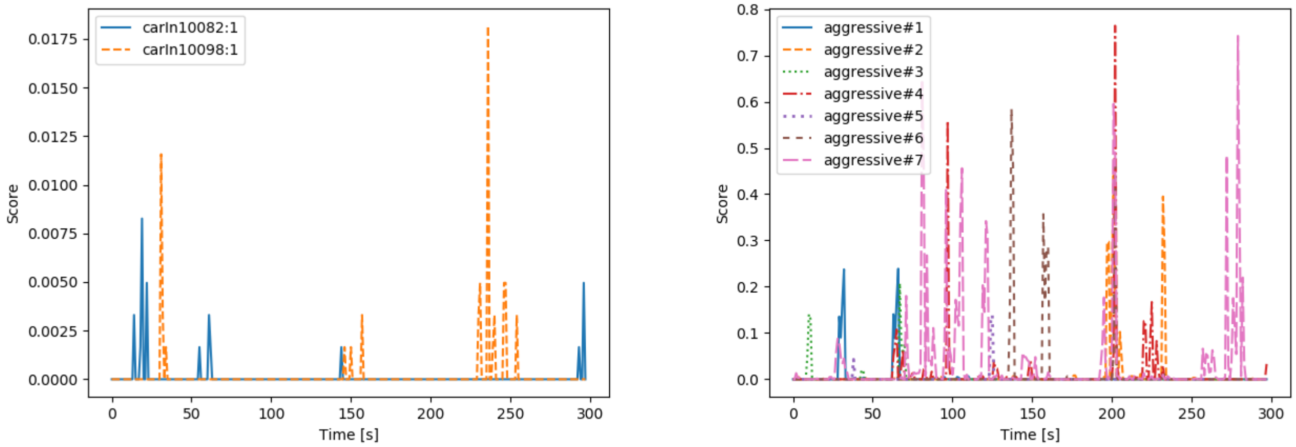
For the SVM, the anomaly score is derived from the sample's distance from the learned decision boundary. k-NN determines the average distance of a sample from its k nearest neighbors and outputs this as anomaly score. In the iForest algorithm, the score is calculated from the average path length that is required to isolate the respective sample with the randomly created decision trees. Here, shorter paths correspond to samples with higher anomaly scores.

D. Post-Processing

For any of the anomaly detection algorithms, a single sample consists of the aggregated values from a time window as described earlier in this section. However, these time windows tend to be quite short (in our case 3 seconds) and thus do not accurately describe the driving behavior of a vehicle over time. Yet, even the most conspicuously behaving vehicles are indistinguishable from normal ones in certain situations — such as when waiting at a traffic light.

To address this, we do not simply use the anomaly scores from the anomaly detectors to decide whether a vehicle behaves abnormally. Instead, we post-process the results by aggregating them over time. This step allows us to draw a conclusion about a vehicle over a certain time frame, such as

[2] http://sumo.dlr.de/wiki/Definition_of_Vehicles,_Vehicle_Types,_and_Routes



(a) Outlier scores of k-NN of exemplary normal vehicles.

(b) Outlier scores of k-NN of exemplary aggressive vehicles.

Fig. 1: Both aggressive and normal vehicles exhibit spikes with higher outlier scores. However, aggressive vehicles have more frequent spikes with significantly larger amplitudes.

e.g., 5 minutes. Observing a vehicle's behavior over a longer time gives us much better results when judging its irregularity.

Post-processing is performed by taking into account all anomaly detection results over some time period for each vehicle. We aggregate these results by taking the maximum, and compare this against a threshold to decide whether a vehicle's behavior is anomalous. In our setting, we chose 5 minutes as a time frame as this gives us sufficient time to reliably detect outliers, it is still short enough for an actual warning system to be useful.

While it is sufficient for us to demonstrate that this approach works, for a deployed system, we would suggest a weighted average that decays over time, since a vehicle that currently behaves anomalously, may very well go back to normal. This also allows to distinguish the intensity of the anomalous behavior by using several thresholds.

IV. Evaluation and Discussion

In this section, we discuss the usage of different features and their impact on detection performance. We also motivate how our post-processing phase improves detection results significantly.

A. Feature Combinations

Out of a wide set of available features, i.e., *acceleration, speeding, lane changes, speed, and minimal gap*, one might think that combining these features altogether will enhance and, hence, deliver the best detection performance. Our evaluation has shown that this is not the case, especially when we consider the distance-based detection algorithms; SVM and k-NN. distance-based algorithms generally have worse results with higher input dimensions, because details are lost due to irrelevant features. Hence, SVM and k-NN do not adapt well to many input dimensions. Furthermore, we find out that the combination of *acceleration and speeding* forms the most promising combination regards SVM and k-NN.

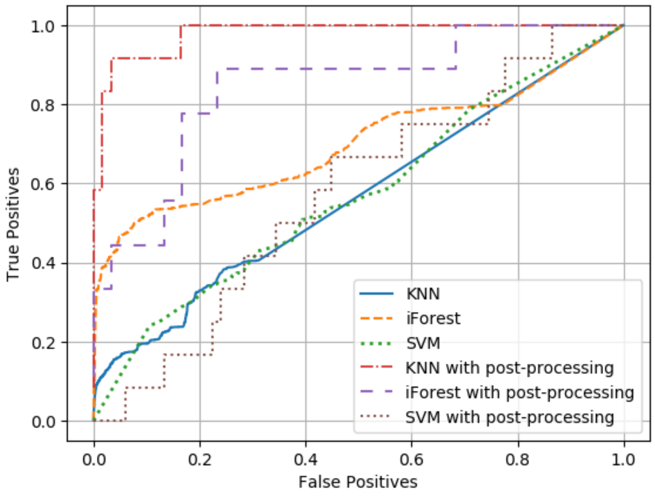


Fig. 2: Receiver operating characteristics for all algorithms without and with post-processing.

B. Changing Vehicle Behavior

As stated earlier, vehicle's behavior can change significantly over time. Drivers who generally exhibit normal behavior can sometimes have short periods of time during which they behave relatively aggressive, or passive. Conversely, abnormally behaving vehicles can seem quite normal for a relatively long time e.g., when they are waiting at traffic lights, or traveling on light-traffic roads.

This can be observed by examining how the outlier detection results develop over time, as shown in Fig. 1. Our post-processing takes this into account by averaging the scores over

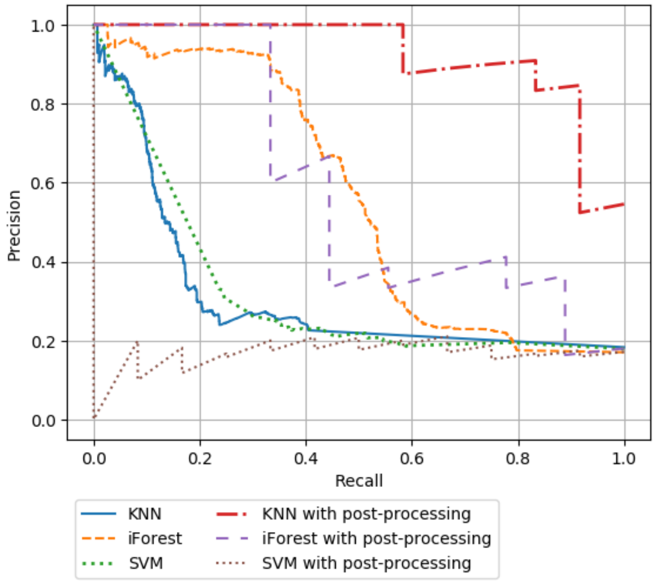


Fig. 3: Precision-recall curves for all algorithms without and with post-processing.

a 5-minute time frame, thus allowing for sufficient time to detect vehicles that exhibit abnormal behavior more often.

C. Prediction Performance

To evaluate the performance, we employ the Receiver Operating Characteristic (ROC), which relates the results' true positive rate against its false positive rate. On the one hand, we require a high true positive rate, as otherwise our system would not detect abnormal vehicles that pose possible threats to traffic safety. On the other hand, the false positive rate should be very low, because false warnings would unnecessarily disturb traffic, or have the effect that the receivers of the warnings pay less attention to them if these are frequently wrong. We show, in Fig. 2, that achieving high detection rates with a low number of false positives is possible by choosing an appropriate threshold, and perform a proper post-processing step which has the effect of increasing the overall accuracy dramatically for both the k-NN and iForest results. The SVM did not perform to a satisfying degree.

Relying on ROC only for evaluation purposes might be misleading, specially when having classes of different sample sizes. Thus, we also show the *precision* against the *recall* plot in Fig. 3. The precision denotes the rate of true positives from all samples that have been flagged as abnormal (true positives & false positives), hence providing a notion of how many of the flagged vehicles are flagged correctly. Moreover, recall considers the true positives from all actual outliers, and gives us an idea on how many abnormal vehicles were missed.

Generally, having a high precision as well as a high recall indicates the best performance of such an anomaly detection system. Fig. 3 shows that we achieve this especially with the combination of using the k-NN detector and our post-processing. However, the tradeoff between the two values has to be considered. We suggest aiming for a higher precision, as false positives may have considerable negative effects, as explained earlier.

V. Conclusion

In this paper, we presented our results of detecting anomalous driving behavior by using common outlier detection mechanisms and post-processing the results with an averaging step.

In the future, we would like to extend our research to not only simulated data, but real-world driving behavior datasets. Furthermore, our goal is to apply and evaluate additional detection mechanisms, such as outlier detection with neural networks.

We tested k-Nearest Neighbors (k-NN), Support Vector Machine (SVM) and Isolation Forest (iForest). While the SVM did not produce usable results, both k-NN and iForest performed very well with high detection rates at low false positive rates. Specifically k-NN with our post-processing proved highly reliable in detecting abnormal vehicles.

References

[1] L. Codeca, R. Frank, and T. Engel, "Luxembourg sumo traffic (lust) scenario: 24 hours of mobility for vehicular networking research," in *IEEE Vehicular Networking Conference (VNC)*, 2015.

[2] M. Amer and M. Goldstein, "Nearest-neighbor and clustering based anomaly detection algorithms for rapidminer," in *Proc. of the 3rd RapidMiner Community Meeting and Conference (RCOMM 2012)*, 2012, pp. 1–12.

[3] L. Xu, K. Crammer, and D. Schuurmans, "Robust support vector machine training via convex outlier ablation," in *AAAI*, vol. 6, 2006, pp. 536–542.

[4] M. Amer, M. Goldstein, and S. Abdennadher, "Enhancing one-class support vector machines for unsupervised anomaly detection," in *Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description*. ACM, 2013, pp. 8–15.

[5] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*, Dec 2008, pp. 413–422.

[6] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation-based anomaly detection," *ACM Trans. Knowl. Discov. Data*, vol. 6, no. 1, pp. 3:1–3:39, Mar. 2012.

[7] M. Enev, A. Takakuwa, K. Koscher, and T. Kohno, "Automobile driver fingerprinting," *Proceedings on Privacy Enhancing Technologies*, vol. 2016, no. 1, pp. 34–50, 2016.

[8] M. Van Ly, S. Martin, and M. M. Trivedi, "Driver classification and driving style recognition using inertial sensors," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 1040–1045.

[9] C. Saiprasert and S. Thajchayapong, "Remote driver identification using minimal sensory data," *IEEE Communications Letters*, vol. 19, no. 10, pp. 1706–1709, 2015.

[10] W. Wang and J. Xi, "A rapid pattern-recognition method for driving styles using clustering-based support vector machines," in *2016 American Control Conference (ACC)*, July 2016, pp. 5270–5275.

[11] M. Goldstein, "Anomaly detection in large datasets," Ph.D. dissertation, Technische Universität Kaiserslautern, Feb. 2014.