Enhancing Road Safety with In-Vehicle Network Abnormal Driving Behavior Detection

Md Rezanur Islam[1], Kamronbek Yusupov[1], Munkhdelgerekh Batzorig[2], Insu Oh[2], and Kangbin Yim[2(✉)]

[1] Department of Software Convergence, Soonchunhyang University, Asan, Korea
{arupreza,yuskamron}@sch.ac.kr
[2] Department of Information Security Engineering, Soonchunhyang University, Asan, Korea
{munkhdelgerekh,catalyst32,yim}@sch.ac.kr

Abstract. This study delves into leveraging Controller Area Network (CAN) data to detect and analyze abnormal driving patterns, underlining its significant role in bolstering road safety measures. By meticulously examining the comprehensive data supplied by the CAN system, which encapsulates real-time inputs from many vehicle sensors and mechanisms, this research marks a pivotal stride in the domain of vehicular safety and intelligent transport networks. The investigation elucidates on categorizing three specific types of unusual driving conduct, showcasing the accuracy and dependability of utilizing CAN data for such purposes. This methodology is a critical breakthrough in crafting instantaneous monitoring systems for erratic driving behavior, aiming to foster safer driving environments.

1 Introduction

Detecting abnormal driving behavior is critical in enhancing road safety and optimizing the efficiency of transportation systems. In recent years, advancements in vehicle technology have opened up new possibilities for analyzing driver behavior using in-vehicle network Controller Area Network (CAN) data [1]. This data, which includes information from various sensors and vehicle components, offers a rich source of insights into how drivers operate their vehicles [2, 3]. Abnormal driving behavior can manifest in various ways, including aggressive driving, distracted driving, reckless maneuvers, mistakenly keeping the door open, and belt off driving, all of which pose significant risks on the road. Detecting and mitigating these behaviors in real time is essential to prevent accidents and improve traffic flow. Understanding driver behavior at a granular level, from individual driving actions to broader patterns, is crucial for achieving these goals. Traditional traffic enforcement and monitoring methods often fail to catch up on detecting subtle or nuanced abnormal driving behaviors that may not immediately lead to accidents or violations [4]. Leveraging CAN data allows for a more comprehensive and nuanced analysis of driver behavior, enabling the identification of deviations from typical driving patterns. This study presents a novel approach to detecting abnormal driver behavior using in-vehicle network CAN data. CAN is a medium that connects Electronic Control Units (ECUs) [5], each responsible for various functions, and facilitates the exchange

L. Barolli (Ed.): IMIS 2024, LNDECT 214, pp. 79–88, 2024.
https://doi.org/10.1007/978-3-031-64766-6_9

of data internally based on the driver's actions. The wealth of information provided by CAN data is leveraged to create a robust system capable of real-time detection and analysis of abnormal driving actions. The system identifies individual driving maneuvers and considers broader patterns and trends in driver behavior. By harnessing the power of in-vehicle network data, the aim is to contribute to developing more effective and data driven solutions for enhancing road safety. The approach offers the potential to create intelligent systems that can alert drivers, provide feedback, and prevent accidents caused by abnormal driving behavior. Through extensive experiments and analysis, the effectiveness and practicality of the method in real-world driving scenarios are demonstrated and presented in Fig. 1.

2 Related Work

Numerous approaches have been developed for detecting abnormal behavior, particularly in various applications such as driver monitoring, vehicle tracking, and safety. These approaches predominantly rely on external data sources, such as images of the driver, images of the moving vehicle, GPS data, physical attributes of the driver, and external vehicle data, including speed, throttle position, and brake pressure. This sector must still be a mountable system despite the many methods employed. In a paper by Wang et al. [6], a hierarchical deep learning approach is employed to classify driving maneuvers and identify drivers' behavior using large-scale GPS sensor data. This involves integrating Deep Neural Networks (DNN) and Long Short-Term Memory (LSTM) networks, preprocessing GPS data, constructing a joint-histogram feature map for regularization, and applying DNN. The model achieved over 94% accuracy in maneuver classification and 92% in driver identification, presenting significant advantages such as high accuracy and detailed characterization of driving behaviors. However, the approach suffers from complexities in implementation, computational intensity, and decreased LSTM performance with increased data scale. Kamaruddin et al. [7] employ speech emotion recognition for analyzing driver behaviors using Mel Frequency Cepstral Coefficients (MFCC) from datasets like the Real-time Speech Driving Dataset (RtSD) and the Berlin Emotional Speech Database (Emo-db). Using classifiers like Multi-Layer Perceptron (MLP), Adaptive Neuro-Fuzzy Inference System (ANFIS), and Generic Self-organizing Fuzzy Neural Network (GenSoFNN), they achieve varying success rates, notably detecting sleepiness with at least 65% accuracy across models. The approach holds potential for real-time applications in vehicle safety systems by alerting drivers of risky behaviors like sleepiness or aggression. However, the moderate overall accuracy, reliance on precise speech data, and complexity in implementation indicate the need for further refinement and research for practical application. In another study [8], the authors employ a deep learning technique, specifically Stacked Denoising Sparse Autoencoders (SdsAEs), to detect abnormal driving behaviors using driving behavior data features such as vehicle speed, throttle position, and brake pressure, normalized against a virtual driver model. The model achieves high micro-recall and micro-precision rates of 92.66% and 92.69%, respectively, demonstrating its effectiveness in distinguishing between everyday driving and four abnormal behaviors: drunk/fatigued, reckless, and phone use. However, the approach's complexity and the need for substantial and diverse data for training are significant challenges. Additionally, the model's reliance on specific features may limit its

applicability across driving contexts or vehicles with varying sensor capabilities. Radtke et al. [9] discuss an algorithm for abnormal driving behavior detection using Adversarial Inverse Reinforcement Learning (AIRL), combining Generative Adversarial Networks (GAN) with Proximal Policy Optimization (PPO) in a multi-agent setting. This approach allows for interaction-aware prediction and more accurate driver behavior modeling. The highD dataset, a top view of vehicle movement collected from drones, is utilized for training and testing. The paper demonstrates that this method outperforms rule-based models and achieves comparable accuracy to state-of-the-art methods with significantly lower inference time. Finally, Shahverdy et al. [10] propose classifying driver behaviors into five distinct categories using Convolutional Neural Networks (CNN). The system collects vehicle data such as acceleration, gravity, RPM, speed, and throttle via a smartphone and an OBD-II adapter for ECU. The data is transformed into images using recurrence plots to capture spatial dependencies, which the CNN classifies into safe, aggressive, distracted, drowsy, or drunk driving behaviors. The approach effectively achieves high classification accuracy, offering an efficient and nonintrusive mechanism for behavior detection. However, it demands considerable computational resources for implementation and operation, relies heavily on the quality of input data, and might face challenges adapting to varying real-world driving scenarios. Despite these challenges, the method significantly advances in applying deep learning to improve road safety and intelligent transportation systems.

Many studies focus on detecting abnormal driver behavior using different data sources, but they often overlook the potential of using CAN data. However, CAN data has the potential to achieve greater precision and reliability in detecting abnormal behavior among drivers. The uniqueness of CAN data lies in its transmission mechanism. It is generated and transmitted based on the actions of ECUs within the vehicle, which govern the vehicle's internal functions that are influenced by the driver's actions. CAN data transmission is situation specific, delivering payload data tailored to specific driving scenarios facilitated by DBC (CAN Database) files [11]. This provides a more direct and fine-grained insight into driver behavior because it reflects the internal workings of the vehicle as influenced by the driver's inputs. Leveraging CAN data can enhance the precision and reliability of abnormal behavior detection, making it a valuable asset in improving road safety and driving behavior analysis.

3 Paper Preparation

3.1 Experimental Setup

In vehicle data collection, researchers typically utilize the On-Board Diagnostics II (OBD-II) system for diagnosis related data collection [12]. However, it should be noted that not all Electronic Control Units (ECUs) are connected to the OBD-II port [13], limiting the types of data that can be collected. To overcome this limitation, the research employs the ECU Direct Approach (EDA), in which data is collected through an internal gateway shown in Fig. 1. The method of line tapping tools is utilized to gather raw data, with an integrated central control unit (ICU) serving as the device that can access the CAN network within the vehicle [13]. The PEAK CAN system [PEAK] [14] is an interfacing device. For data analysis, Python Jupyter [15] and Pytorch for deep learning [16] are

employed. The experiments are conducted on a system equipped with an Intel(R) Core (TM) i9-10900K CPU @ 3.70 GHz and an NVIDIA GeForce RTX 2080 super graphics card.

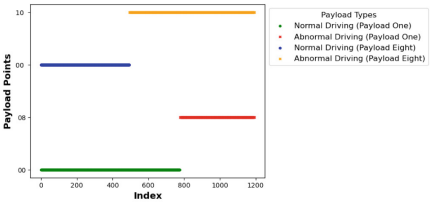
Fig. 1. Data Collection Through CAN Gateway [13]

3.2 Feature Extraction

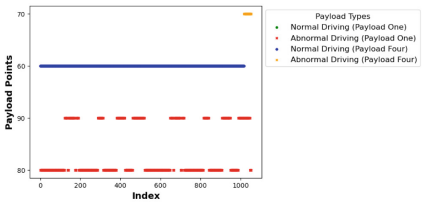
The experimental process commences with data collection, involving the selection of a specific route. Data collection occurs in two stages: firstly, during normal and secure driving conditions, and secondly, during the collection of data containing abnormal activities. Subsequently, the data analysis phase is initiated. The data analysis technique hinges on identifying differences in the payload for entries sharing the same ID. Given that CAN data comprises eight-byte hexadecimal payloads, the analysis is performed by systematically examining each byte's position, one by one.

This Algorithm 1 is designed to compare two DataFrames, df1 and df2, with a list of specific IDs and divide the data into two DataFrames: one containing rows with matching IDs and another containing rows with non-matching IDs. The algorithm initializes two empty lists, match ids and non-match ids, to store the results. It iterates through each ID (id) in the given list of IDs. For each ID, it filters the rows in df1 and df2 based on the 'CAN ID' column to create df1 i and df2 i, which are subsets of the original DataFrames. Then, the algorithm calls the function Algorithm 2 to compare the data in df1 i and df2 i. This function should return two DataFrames: one with matching rows (match) and one with non-matching rows (non-match). The matching and non-matching rows are appended to match ids and non-match ids, respectively. After processing all the IDs, the algorithm converts match ids and non-match ids into DataFrames and removes any rows with missing values. Finally, it returns the two DataFrames: match ids containing rows with matching IDs and non-match ids containing rows with non-matching IDs. In summary, this algorithm systematically compares two DataFrames based on specific IDs, enabling the separation of data into matching and nonmatching categories for further analysis.

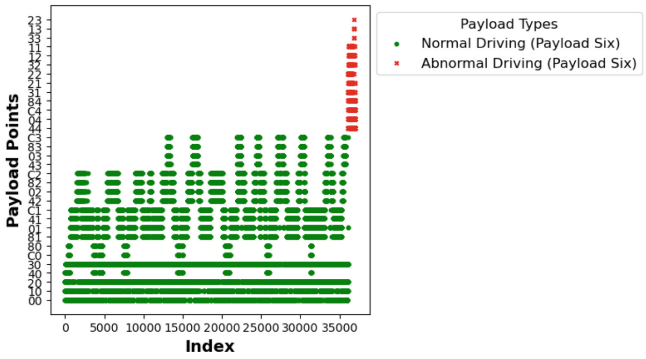
This Algorithm 2 is designed to compare two payload DataFrames, denoted as df1 and df2, and identify the differences in their byte values. It aims to find unique values in each column of df1 that are not present in the corresponding columns of df2. The



(a) Payload Distribution Under Door Abnormality



(b) Payload Distribution Under Belt Abnormality



(c) Payload Distribution Under Break Abnormality

Fig. 2. Payload Distributions Under Different Abnormalities

algorithm initializes an empty list called out to store the results. It then iterates through each column (col) in df1. For each column, it extracts the unique values (unique values df1) found in that column in df1, as well as the unique values (unique values df2) in the corresponding column of df2. Next, the algorithm enters a nested loop to examine each value (val) in unique values df1. It checks whether each val is present in unique values df2. If val is not found in unique values df2, it appends a tuple consisting of the column name (col) and the unique value (val) to the out list. Finally, the algorithm returns the out list, which contains all the column names and values from df1 that are not present in the corresponding columns of df2. In summary, this algorithm systematically identifies and collects the discrepancies between two DataFrames, providing a clear and structured approach to pinpointing where and what values differ between the two datasets.

In this study, the primary focus is on detecting three specific abnormal behaviors: door abnormalities, belt abnormalities, and brake abnormalities. The main emphasis is identifying brake abnormalities and investigating door and belt abnormalities, as they are readily available in current vehicles. This approach allows for the evaluation of the performance of the analysis algorithms (Algorithm 1 and 2).

Specific CAN IDs associated with these abnormal behaviors have been identified through the analysis. CAN ID 0x018 corresponds to door and belt functions, while CAN IDs 0x4B0 and 0x220 are responsible for brake related data. ECUs transmit data over the CAN network based on interdependent functions. For instance, the RPM, speed,

and brake functions are interconnected. When the brake is engaged, the RPM decreases, and the speed also decreases. Consequently, a sequential impact is observed in the CAN data, particularly in the case of brake-related actions. However, door and belt functions exhibit different interdependencies. In the case of interdependent functions, substantial changes are noticeable in the data. Figure 2c demonstrates that under brake abnormality conditions, the number of unique payload values is higher compared to door and belt abnormalities shown in Fig. 2a and b. Only byte positions one and four are affected for door abnormalities, while for belt abnormalities, it is byte positions one and eight. In contrast, for brake abnormalities, except for byte positions two, four, six, and eight, the other positions remain relatively unchanged. Therefore, the data analysis is structured based on the payload values, considering these distinct characteristics. This approach ensures a comprehensive and nuanced examination of the data, enabling effective detection and differentiation between these abnormal driving behaviors.

Table 1. Deep Learning Hyperparameters.

Hyperparameters	Value
Batch size	64
Hidden size	64
Output size	4
Learning Rate	0.001
Loss Function	CrossEntropy
Optimizer	Adam
Early Stopping	10
Best Accuracy	0.9607
Counter	33

3.3 Deep Learning Model

In this study, a deep learning model was developed to classify abnormal driving behavior. The LSTM-based classifier was designed to analyze complex data sequences derived from various vehicle sensors. The model architecture comprises an LSTM layer and a fully connected layer for classification. The hyperparameters of the model as shown in Table 1, including batch size, hidden size, and learning rate, were tailored to optimize its performance. A CrossEntropy loss function and the Adam optimizer were also employed for training. An early stopping mechanism was incorporated to ensure efficient training and prevent overfitting, monitoring the model's accuracy over epochs. The experiments yielded promising results, with the best accuracy of 96.07% achieved after 33 training epochs. This research significantly contributes to the development of advanced systems for real-time detection of abnormal driving behavior, potentially enhancing road safety and traffic management.

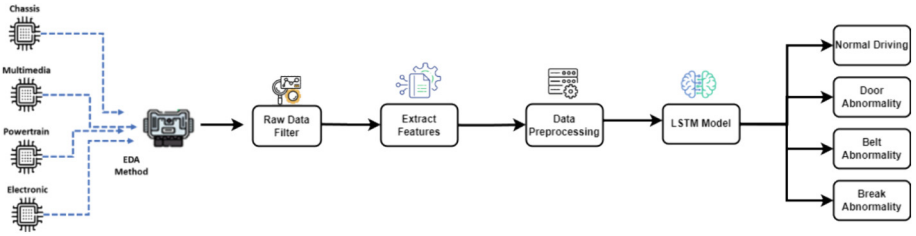


Fig. 3. Comprehensive Working Flow of Abnormal Behave Detection

4 Result Evaluation and Discussion

4.1 Working Flow

The workflow for a vehicle monitoring system that employs an EDA method to process and analyze data from various car subsystems, such as Chassis, Multimedia, Powertrain, and Electronics, is as follows: The data collected from these subsystems is initially passed through a 'Raw Data Filter' to eliminate noise and irrelevant information as shown in Fig. 3. Following this initial filtration, the data undergoes a 'Feature Extraction' process in which specific characteristics are identified and extracted for further analysis. After the relevant features have been extracted, the data is preprocessed to prepare it for the final analysis phase. This preprocessing may include normalization, scaling, or data transformation to ensure it is in a suitable format for machine learning models. The preprocessed data is then input into an LSTM model, a recurrent neural network (RNN) well-suited for time-series data. The LSTM model's responsibility is to identify patterns and abnormal behaviors in the data, which could indicate different vehicle operation states, such as 'Normal Driving,' 'Door Abnormality,' 'Belt Abnormality,' or 'Brake Abnormality.' Each of these outcomes is crucial for ensuring the vehicle's safe operation and providing actionable insights that can lead to maintenance or immediate corrective actions if necessary.

4.2 Result Evaluation

In the result evaluation section, the performance of the classification model was analyzed through various metrics. The confusion matrix offers a detailed view of the model's predictions, showing a high accuracy level in identifying 'Normal', 'Door Abnormality', 'Belt Abnormality' classes with no false positives or negatives. 'Break Abnormality' exhibited some misclassifications with 'Normal', highlighted by an 11.93% error rate in this category. The model achieved a notable 96.33% as shown in Table 2, accuracy in predicting the 'Normal' class and a perfect 100% in detecting 'Door' and 'Belt' abnormalities.

The Receiver Operating Characteristic (ROC) curve reinforces these results, showing Area Under the Curve (AUC) scores near perfection for 'Door Abnormality' and 'Belt Abnormality' at 1.00 and slightly lower for 'Normal' and 'Break Abnormality' at 0.97 and 0.94 respectively. This indicates a remarkable discriminative capability of the model across all classes. The classification report reveals high precision and recall for all classes,

resulting in F1 scores demonstrating a well-balanced model. The 'Door Abnormality' and 'Belt Abnormality' classes achieved perfect precision and recall, achieving an F1-score of 1.00. The 'Normal' and 'Break Abnormality' classes also showed high precision and recall, with F1 scores of 0.96 and 0.90, respectively.

The model's overall accuracy stands at an impressive 97%, with macro and weighted averages across precision, recall, and F1-score of 0.97, signifying the model's robustness. The error rate is low at approximately 3.46%, and the false alarm rate is non-existent, indicating the model's dependability in practical scenarios. The ROC AUC score of 0.9768 further confirms the strong performance of the model across all classes as shown in Figs. 4 and 5. These metrics collectively validate the classification model's effectiveness in differentiating between classes with high reliability and minimal error, making it a valuable tool for its intended application domain.

Table 2. Classification Report and Performance Metrics

Class	Precision	Recall	F1-score
Normal	0.96	0.97	0.96
Door Abnormality	1.00	1.00	1.00
Belt Abnormality	1.00	1.00	1.00
Break Abnormality	0.91	0.89	0.90
Macro Avg	0.97	0.96	0.97
Weighted Avg	0.97	0.97	0.97
Error Rate	0.0346		
False Alarm Rate	0.0000		
ROC AUC Score	0.9768		
Accuracy	0.97		

4.3 Discussion and Limitation

The research presented here highlights CAN data's significant yet underutilized potential in analyzing abnormal driving behaviors. CAN data is renowned for its accuracy and reliability and is an optimal solution in this sector. However, its complexity poses a challenge. The intricate nature of in-vehicle networks, characterized by numerous ECUs and the voluminous generation of data, coupled with the dynamic characteristics of CAN data, makes analysis intricate. In this study, three types of abnormal behaviors have been successfully classified. However, identifying and categorizing various types of abnormal driving behaviors to develop a comprehensive abnormal behavior detection system requires an extensive survey. A pivotal in this process is the CAN DBC file, which serves as the blueprint for CAN network communications. Utilizing the CAN DBC could significantly enhance the accuracy and reliability of the system. Nevertheless, the proprietary nature of CAN DBC files, typically held confidential by manufacturers,

presents a barrier. The future objective involves adopting a reverse engineering approach to the CAN DBC. This strategy aims at extracting relevant features in conjunction with a detailed survey of abnormal driving behaviors. By doing so, the driving safety field can be advanced, implementing an effective system for detecting abnormal driving behaviors. This endeavor could lead to more secure and safer driving environments.

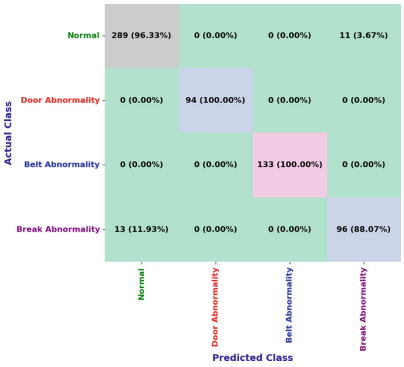


Fig. 4. Confusion Matrix in the Classification Report for Abnormal Driving Behavior Detection

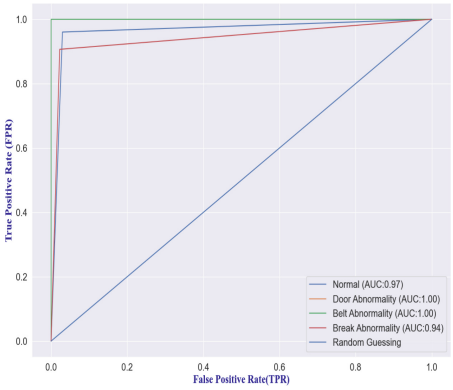


Fig. 5. ROC Curve for Abnormal Driving Behavior Detection

5 Conclusion

The study highlights the potential of utilizing CAN data to pinpoint abnormal driving behaviors. The researchers successfully classified three distinct behaviors, showing the effectiveness and precision of using CAN data. These findings emphasize the valuable insights that CAN data provides into the intricate workings of vehicles influenced by driver actions. Consequently, this approach represents a highly promising avenue to enhance road safety. This research contributes significantly to intelligent transportation systems, laying the groundwork for developing more advanced methods to detect abnormal driving behaviors. Ultimately, these advancements aim to create safer roads for everyone.

Acknowledgments. This work was supported by Institute for Information and communications Technology Planning and Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2022-0-01197, Convergence security core talent training business (Soon Chun Hyang University)) and supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A4A2001810).

References

1. Azadani, M.N., Boukerche, A.: Driving behavior analysis guidelines for intelligent transportation systems. IEEE Trans. Intell. Transp. Syst. **23**(7), 6027–6045 (2021)

2. Abdennour, N., Ouni, T., Amor, N.B.: Driver identification using only the CAN-Bus vehicle data through an RCN deep learning approach. Robot. Auton. Syst. **136**, 103707 (2021)
3. Gazdag, A., Lestyán, S., Remeli, M., Ács, G., Holczer, T., Biczók, G.: Privacy pitfalls of releasing in-vehicle network data. Veh. Commun. **39**, 100565 (2023)
4. Khan, K., Zaidi, S.B., Ali, A.: Evaluating the nature of distractive driving factors towards road traffic accident. Civ. Eng. J. **6**(8), 1555–1580 (2020)
5. Islam, M.R., Sahlabadi, M., Kim, K., Kim, Y., Yim, K.: CF-AIDS: Comprehensive Frequency-Agnostic Intrusion Detection System on In-Vehicle Network. IEEE Access (2023)
6. Wang, Y., Ho, I.W.-H.: Joint deep neural network modelling and statistical analysis on characterizing driving behaviors. In: 2018 IEEE Intelligent Vehicles Symposium (IV), pp. 1–6. IEEE (2018)
7. Kamaruddin, N., Wahab, A.: Driver behavior analysis through speech emotion understanding. In: 2010 IEEE Intelligent vehicles symposium, pp. 238–243. IEEE (2010)
8. Hu, J., Zhang, X., Maybank, S.: Abnormal driving detection with normalized driving behavior data: a deep learning approach. IEEE Trans. Veh. Technol. **69**(7), 6943–6951 (2020)
9. Radtke, H., Bey, H., Sackmann, M., Schön, T.: Predicting driver behavior on the highway with multi-agent adversarial inverse reinforcement learning. In: 2023 IEEE Intelligent Vehicles Symposium (IV), pp. 1–8. IEEE (2023)
10. Shahverdy, M., Fathy, M., Berangi, R., Sabokrou, M.: Driver behavior detection and classification using deep convolutional neural networks. Expert Syst. Appl. **149**, 113240 (2020)
11. Hoang, T.N., Islam, M.R., Yim, K., Kim, D.: CANPerFL: improve in-vehicle intrusion detection performance by sharing knowledge. Appl. Sci. **13**(11), 6369 (2023)
12. Rimpas, D., Papadakis, A., Samarakou, M.: OBD-II sensor diagnostics for monitoring vehicle operation and consumption. Energy Rep. **6**, 55–63 (2020)
13. Koh, Y., Kim, S., Kim, Y., Oh, I., Yim, K.: Efficient CAN dataset collection method for accurate security threat analysis on vehicle internal network. In: Barolli, L. (ed.) Innovative Mobile and Internet Services in Ubiquitous Computing: Proceedings of the 16th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2022), pp. 97–107. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-031-08819-3_10
14. Plšičík, R., Danko, M.: API for data transfer using USB to CAN converter. In: 2023 IEEE 32nd International Symposium on Industrial Electronics (ISIE), pp. 1–6. IEEE (2023)
15. Islam, M.R., Oh, I., Yim, K.: CANTool an in-vehicle network data analyzer. In 2022 International Conference on Information Technology Systems and Innovation (ICITSI), pp. 252–257. IEEE (2022)
16. Ketkar, N., Moolayil, J.: Introduction to pytorch. In: Ketkar, N., Moolayil, J. (eds.) Deep learning with python: learn best practices of deep learning models with PyTorch, pp. 27–91. Apress, Berkeley, CA (2021). https://doi.org/10.1007/978-1-4842-5364-9_2